Modelling of Intrusion Detection System Using Artificial Intelligence—Evaluation of Performance Measures

Manojit Chattopadhyay

Abstract In recent years, applications of internet and computers are growing extremely used by many people all over the globe—so is the susceptibility of the network. In contrast, network intrusion and information security problems are consequence of internet application. The increasing network intrusions have placed people and organizations to a great extent at peril of many kinds of loss. With the aim to produce effectiveness and state-of-the-art concern, the majority organizations put their applications and service things on internet. The organizations are even investing huge money to care for their susceptible data from diverse attacks that they face. Intrusion detection system is a significant constituent to protect such information systems. A state-of-the-art review of the applications of neural network to Intrusion Detection System has been presented that reveals the positive trend towards applications of artificial neural network. Various other parameters have been selected to explore for a theoretical construct and identifying trends of ANN applications to IDS. The research also proposed an architecture based on Multi Layer Perceptron (MLP) neural network to develop IDS applied on KDD99 data set. Based on the identified patterns, the architecture recognized attacks in the datasets using the back propagation neural network algorithm. The proposed MLP neural network has been found to be superior when compared with Recurrent and PCA neural network based on the common measures of performance. The proposed neural network approach has resulted with higher detection rate (99.10 %), accuracy rate (98.89 %) and a reduced amount of execution time (11.969 s) and outperforms the benchmark results of six approaches from literature. Thus the analysis based on experimental outcomes of the MLP approach has established the robustness, effectiveness in detecting intrusion that can further improve the performance by reducing the computational cost without obvious deterioration of detection performances.

Keywords Artificial intelligence · Multilayer perceptron · Intrusion detection system · Detection rate · Fasle alarm · KDDCUP99

M. Chattopadhyay (✉)
Operations and Systems Area, Indian Institute of Management Raipur, GEC Campus,
Sejbahar, Raipur 492015, Chhattisgarh, India
e-mail: mjc02@rediffmail.com

Q. Zhu and A.T. Azar (eds.), *Complex System Modelling and Control Through Intelligent Soft Computations*, Studies in Fuzziness and Soft Computing 319,
DOI 10.1007/978-3-319-12883-2_11

1 Introduction

In recent years, intrusion detection system (IDS) has attracted a great deal of concern and attention. The webopedia English Dictionary (http://www.webopedia.com/) defines intrusion detection system as "An intrusion detection system (IDS) inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system." (Heady et al. 1990). Heady et al. (1990) describe intrusion as "any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource". Even after adopting various intrusion prevention techniques it is nearly impossible for an operational system to be completely secure (Lee et al. 1999). Therefore IDS are imperative to provide extra protection for being characterized as normal or legitimate behaviour of resources, models and techniques rather than to identify as abnormal or intrusive. The IDS has been formalized during the 1980s as a potential model (Denning 1987) to prevent the incident of unauthorized access to data (Eskin et al. 2002). During the last two decades has been categorized accepted definition of financial fraud, Wang et al. (2006) define it as "a deliberate act that is contrary to law, rule, or policy with intent to obtain unauthorized financial benefit."

Therefore due to the immense expansion of computer networks usage and the enormous increase in the number of applications running on top of it, network security is becoming more and more significant. As network attacks have increased in number and severity over the past few years, consequently Intrusion Detection Systems (IDSs) is becoming more important to detect anomalies and attacks in the network. Therefore, even with the most advanced protected environment, computer systems are still not 100 % secure.

In the domain of intrusion detection, there is a growing interest of the application and development of Artificial Intelligence (AI) based approach is (Laskov et al. 2005). AI and machine learning techniques were used to discover the underlying models from a set of training data. Commonly used methods were rule-based induction, classification and data clustering (Wu and Bunzhaf 2010). AI is a huge and sophisticated field still growing and certainly not optimized for network security. Definite effort will be required in AI to help its application to IDSs. Development on that face will take place more rapidly if the opportunity of using AI techniques in IDSs motivates more attention to the AI community. AI is a collection of approaches, which endeavors to make use of tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low solution cost. As AI techniques can also be used for computational intelligence, different computational intelligence approaches have been used for intrusion detection (Fuzzy Logic, Artificial Neural Networks, Genetic Algorithms) (Yao et al. 2005; Gong et al. 2005; Chittur 2001; Pan et al. 2003), but their potentials are still underutilized. Researcher are also using a term computational intelligence that deals with only numerical data to recognize patterns unlike that of artificial intelligence it has potential to computational adaptive, fault tolerant, maximizing speed, minimizing error rates corresponding to human performance (Bezdek 1994).

Wu and Banzhaf (2010) commented that the popular domain of AI is different from the CI. However there is neither full conformity on the exact nature of computational intelligence nor there is any far and wide established vision on which domain belong to CI: artificial neural networks, fuzzy sets, evolutionary computation, artificial immune systems, swarm intelligence, and soft computing. Majority of these approaches are able to process the information using either supervised or unsupervised learning algorithm. Supervised learning frequently constructs classifiers known as a function mapping data observations to matching class labels for misuse discovery from class-labeled training datasets. Classifiers are basically viewed. On the other hand, unsupervised learning is different from supervised learning due to non-availability of class-labeled data during the training stage and it works on based on similarities of data points. Therefore it becomes a more suitable approach to deal with anomaly detection.

Artificial Intelligence (AI) has recently been attracted significantly in the development of Intrusion Detection System (IDS) for anomaly detection, data reduction from the research community. Due to large trend of internet usage in the last decade in a more complex and un-trusted global internet environment, the information systems are inescapably uncovered to the growing threats. Intrusion Detection System is an approach use to respond to such threats. Diverse IDS techniques have been proposed, which identify and alarm for such threats or attacks. The Intrusion Detection System (IDS) generates huge amounts of alerts that are mostly false positives. The abundance of false positive alerts makes it difficult for the security analyst to identify successful attacks and to take remedial actions. Many of artificial intelligence approach have been used for classification, but they alone are incapable of dealing with new types of attack which are evolving due to the advent of real time data. To address with these new problems of networks, artificial intelligence based IDS are opening new research avenues. Artificial intelligence offers a vast range of techniques to classify these attacks. So to assist in categorizing the degree of the threat, different artificial intelligence techniques are used to classify the alerts, our research work will be based on analyzing the existing techniques and in the process identifying the best algorithm for the development of an efficient intrusion detection system.

The fundamental objectives of our contribution will be to explore for an optimal intrusion detection system model based on Artificial Intelligence techniques and evaluation perspective for performance of such predictive classification system. Therefore, the objective is basically to provide solutions in developing a complex system model. The principal chapter objectives of this research work can be summarized as:

1. Undertake detailed study on anomaly based intrusion detection systems.
2. Exploring the research trend for security challenges of ID based on anomaly detection after critical appraisal of the existing methodologies for intrusion detection system.

3. Propose a suitable methodology for anomaly detection using KDD99Cup Dataset. Specifically, the research work focuses on the followings:
   (a) To extract the data, normalize it and categorization of the attack based on numerical value
   (b) to develop an optimal neural network architecture of the anomaly detection for increase rate of correct classification of anomaly
   (c) to calculate the performance measure of the anomaly in IDSs result obtained after applying proposed supervised learning approach
   (d) to assess the predictive ability of the proposed neural network architecture

In the present research the intrusion detection has been considered as a binary classification problem and thus it is necessitated to highlight the back ground on the types of intrusion detection system in the next section.

1.1 Intrusion Detection

Intrusion detection mechanism can be divided into two broad categories (Anderson 1995; Tiwari 2002) (i) Misuse detection system (ii) Anomaly based detection.

The systems are described as below:

(i) Misuse detection system

It is perhaps the oldest and most frequent method and applies well-known knowledge of identified attack patterns to search for signatures, observe state transitions or employed at a mining system to classify potential attacks (Faysel and Haque 2010). The familiar attacks can be identified efficiently with a very low false alarm rate for which it is broadly applied in most of the commercial systems. As the attacks are frequently polymorph, and changed regularly therefore, misuse detection become unsuccessful due to unfamiliar attacks. This problem may be resolved by regularly updated knowledge base either through time consuming and laborious manual method or through automatic updating using supervised learning methods. However this becomes too costly to set up to perform labeling of each occurrence in the dataset as normal or a type of attack. Differently to deal with this problem is to apply the anomaly detection method as proposed by Denning (1987).

(ii) Anomaly based detection

Anomaly detection systems recognize difference from normal behaviour and alert to possible unknown or novel attacks lacking any past knowledge of them. It theorized that anomalous behavior is rare and dissimilar from normal behavior. Thus it is orthogonal to misuse detection (Wu and Banzhaf 2010). Anomaly detection can be of two types (Chebrolu et al. 2005): static and dynamic anomaly detection. In the first one it is assumed that the observed attack behavior is constant and the second one extracts pattern occasionally known as profiles from behavioral routine of end users, or usage history of networks/hosts.

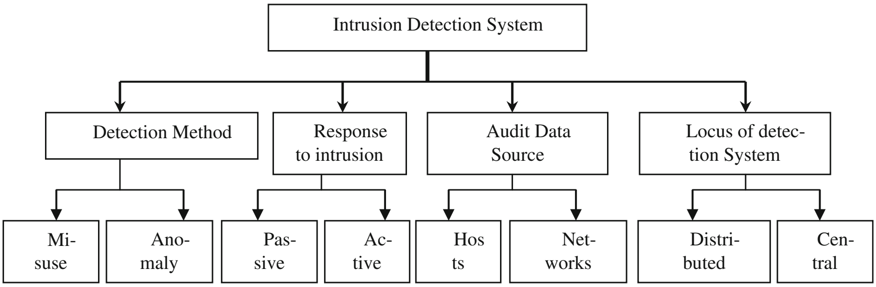


Fig. 1 Classification scheme of intrusion detection system taken from Wu and Banzhaf (2010)

Therefore, anomaly detection has the potential of identifying latest kind of attacks, and only necessitates normal data during generation of profiles. Though, the main intricacy involves in determining borders among normal and abnormal behaviors, as a result of the lack of abnormal examples during the learning stage. An additional complexity is to familiarizing itself to continually varying normal behavior, particularly for dynamic anomaly detection.

Additionally there are other features used to classify intrusion detection system approach, as shown in Fig. 1 (Wu and Banzhaf 2010).

One frequent method applied to identify intrusion detection is by classification defined as dividing the samples into distinct partition. The purpose of the classifier is not to investigate the data to determine interesting partition but also to settle on how new data will be classified. In intrusion detection, classification grouped the data records in a encoded classes applied as features to label each sample, discriminating elements fitting to anomaly or normal attack classes. However the classification has to be used with fine tuning approaches to decrease false positive rates. Thus intrusion detection is considered as a binary categorization problem (Liao and Vemuri 2002).

Artificial neural network is relatively new and emerging approach to easily deal with complex classification with much better precision and output and the conceptual background of different types of artificial neural network with diverse application domains explored in literature are discussed in the next section.

1.2 *Artificial Neural Network*

Artificial intelligence (AI) is an interdisciplinary domain exhibits human-like intelligence and demonstrated by hardware or software. The term AI was coined by McCarthy et al. (1955) and defined it as “the science and engineering of making intelligent machines” (McCarthy 2007). Artificial Neural Network (ANN) is massively parallel interconnections of simple neurons that act as a collective system (Haykin 2005). The ANNs mimic the human brain so as to perform intelligently. The major benefits include high computation rate due to their massive parallelism for which real time computation of large data sets become possible using proper hardware. The information is determined on connection weights between the layers. A processing unit consists of a learning rule and an activation function. The learning rule resolves the actual input of the node by mapping the output of all direct antecedent and extra external inputs onto a single input value. The activation function is then applied on the actual input and determines the output of the node. The output of the processing unit is also described as activation. In the Fig. 2 the two input nodes are shown in input layer, one output nodes is shown in output layer. Organizing the nodes in layers resulted in a layered network and the Fig. 2 shows in between input and output layers there are two hidden layers. The inputs to hidden and hidden to output nodes are connected by weight values that is initialized during the start of the training and a net input is calculated on which the activation function is applied to calculate the output. The multilayer perceptron has additional L ≥ 1 hidden layers. The lth hidden layer consists of h(l) hidden units. MLP is applied to solve wide varieties of interdisciplinary problems like credit scoring (Khashei et al. 2013), medical (Peláez et al. 2014), food classification (Dębska and Guzowska-Świder 2011), forecasting (Valero et al. 2012), mechanical engineering (Hwang et al. 2010), production (Kuo et al. 2010) etc.

The next section will discuss specifically the various neural network approaches applied in the development of intrusion detection system.

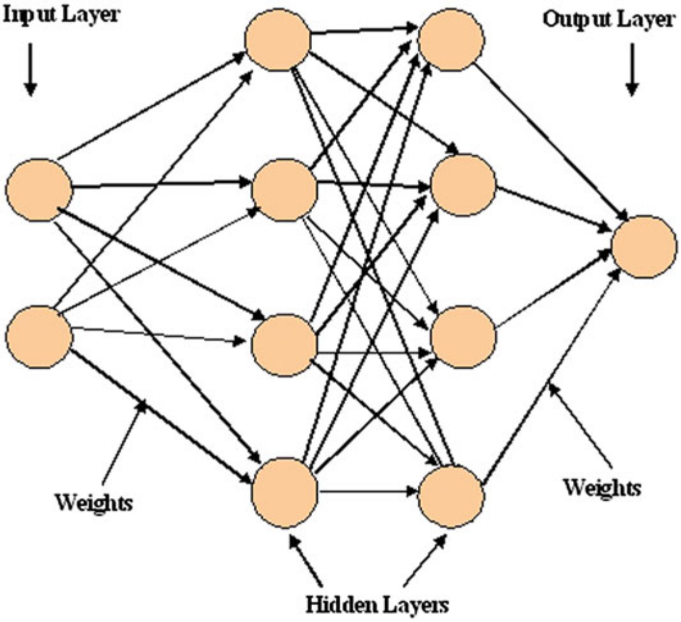


Fig. 2 ANN architecture

2 Survey on the AI Based Techniques Used for Intrusion Detection

Artificial neural network based intrusion detection system development is an important research trend in intrusion detection domain (Yang et al. 2013). Artificial Neural Network (ANN) has been used in the classification process of the system. The inputs of ANN are obtained from the features of packet headers, such as port number and IP number. The implemented embedded IDS has been first trained with training data. Then, packet classification has been performed in the real time and finally time of determining packet classes have been obtained (Tuncer and Tatar 2012). ANN has been shown to increase efficiency, by reducing the fault positive, and detection capabilities by allowing detection with partial available information on the network status (El Kadhi et al. 2012).

Different sizes of feed forward neural networks are compared for their evaluation performance using MSE. The generalization capacity of the trained network shows potential and the network is competent to predict number of zombies involved in a DDoS attack with very less test error (Gupta et al. 2012). Genetic Algorithm has successfully applied on NSL-KDD data set (Aziz et al. 2014). Research has revealed high accuracy and good detection rates but with moderate false alarm on novel attacks by the implementing Genetic Algorithms, Support Vector Machines, Neural Networks etc. (Abdel-Aziz et al. 2013; Zainaddin et al. 2013). In a research it is established that PSO outperforms GA both in population size and number of evolutions and can converge faster. Comparing PSO with some other machine learning algorithm it was found that PSO perform better in terms of detection rate, false alarm rate, and cost per example (Sheikhan and Sharifi 2013).

IDS development using Self Organization Map (SOM) neural network, has been successfully detected anomalies (Xiang et al. 2013). Comparative result analysis of SOM implementation based on several performance metrics revealed that detection rate for KDD 99 dataset was 92.37 %, while detection rate for NSL-KDD dataset was 75.49 % (Ibrahim et al. 2013).

ART2 neural network experiments with IDS demonstrated that the model effectively improved detection accuracy and decreased false alarm rate compared with the static learning intrusion detection method based on SVM (Liu 2013). Fuzzy adaptive resonance theory-based neural network (ARTMAP) has been used as a misuse detector (Sheikhan and Sharifi 2011).

In majority of the research ANNs has improved the performance of intrusion detection systems (IDS) when evaluated with traditional approaches. However for ANN-based IDS, detection precision, especially for low-frequent attacks, and detection stability are still required to be improved. FC-ANN approach, based on ANN and fuzzy clustering, has demonstrated to solve IDS that achieved higher detection rate, less false positive rate and stronger stability. Experimental outcomes on the KDD CUP 1999 dataset showed that FC-ANN approach outperforms BPNN and other well-known approaches like decision tree, the naive Bayes in terms of detection precision and detection stability (Wang et al. 2010).

Recurrent Neural Network out-performs Feed-forward Neural Network, and Elman Network for detecting attacks in a communication network (Anyanwu et al. 2011).

Theory and experiment show that Radial basis function network (RBFN) algorithm has better ability in intrusion detection, and can be used to improve the efficiency of intrusion detection, and reduce the false alarm rate (Peng et al. 2014). Binary Genetic Algorithm (BGA) as a feature extractor provide input for the classification task to a standard Multi-layer Perceptron (MLP) classifier that resulted with very high classification accuracy and low false positive rate with the lowest CPU time (Behjat et al. 2014).

Using k-means clustering, Naive Bayes feature selection and C4.5 decision tree classification for pinpointing cyber attacks resulted with a high degree of accuracy (Louvieris et al. 2013). Comparing the traditional BP networks and the IPSO-BPNN algorithm to simulate results of the KDD99 CUP data set with the intrusion detection system has demonstrated the BPN resulted with less time, better recognition rate and detection rate (Zhao et al. 2013).

Feizollah et al. (2014) evaluated five machine learning classifiers, namely Naive Bayes, k-nearest neighbour, decision tree, multi-layer perceptron, and support vector machine in wireless sensor network (WSN). A critical study has been made using genetic algorithm, artificial immune, and artificial neural network (ANN) based IDSs approaches (Yang et al. 2013).

A network IDS applied discretization with genetic algorithm (GA) as a feature selection to assess it’s performance several classifiers algorithms like rules based classifiers (Ridor, Decision table), trees classifiers (REPTree, C 4.5, Random Forest) and Naïve bays classifier have been used on the NSL-KDD dataset (Aziz et al. 2012; Eid et al. 2013). Research revealed that discretization has a positive impact on the time to classify the test instances and is found to be an important factor for developing a real time network IDS.

Therefore only a detail analytical view on applications of neural network based intrusion detection system can quantitatively enlighten on the trend of kind of diverse research based on neural network as explored in literature.

2.1 Analysis of IDS Research Based on the Neural Network Algorithm

This paper provides a state-of-the-art review of the applications of neural network to IDS. The following query string has been searched using scopus search engine: (TITLE-ABS-KEY (intrusion detection system) AND SUBJAREA (mult OR ceng OR CHEM OR comp OR eart OR ener OR engi OR envi OR mate OR math OR phys) AND PUBYEAR > 1999) AND (neural network). It resulted with 2,185 articles and only the relevant information has been collected to interpret the significance of IDS research using neural network during the period 2000–2014. Figures 3, 4, 5, 6 and 7 organizes this review of the literature.

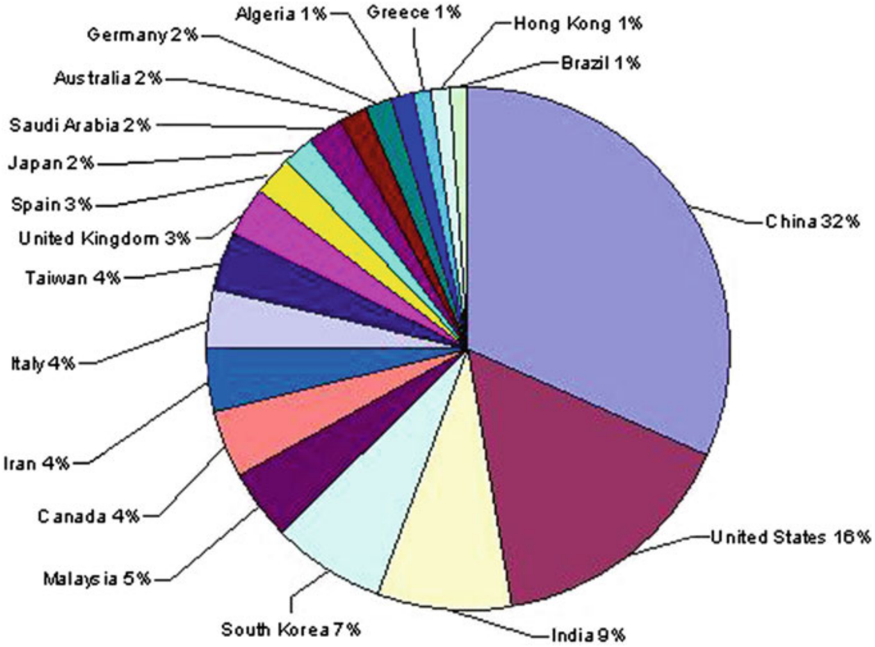


Fig. 3 Articles on neural network applied intrusion detection system development published by researcher from their affiliated country

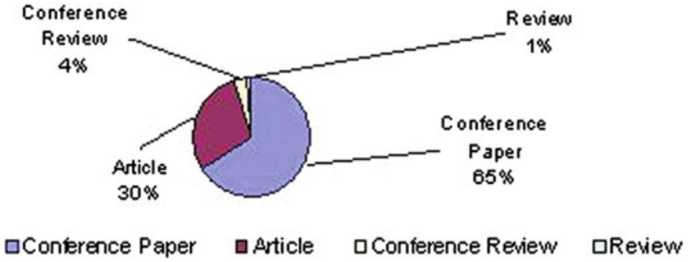


Fig. 4 Type of research documents published on neural network applied intrusion detection system development

Figures 3, 4, 5 and 6, dissects and organizes this review of the literature. For the classification of literature Fig. 3 shows the articles published by researcher from their affiliated country. It is shown in that China is leading (32 %) followed by USA (16 %) and India (9 %) as highest articles published by affiliated country.

The conference papers (65 %) are the major type of research documents followed by articles (30 %) as revealed by Fig. 4.

The Fig. 5 has not considered around 329 articles published in rest 141 journals having less than 7 articles published due to interpretability of this huge in formation

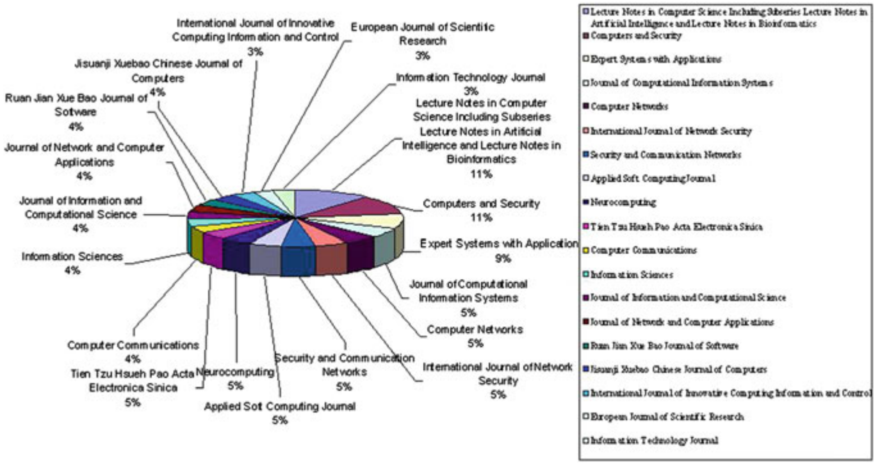


Fig. 5 Major journals published articles related to neural network applied intrusion detection system development

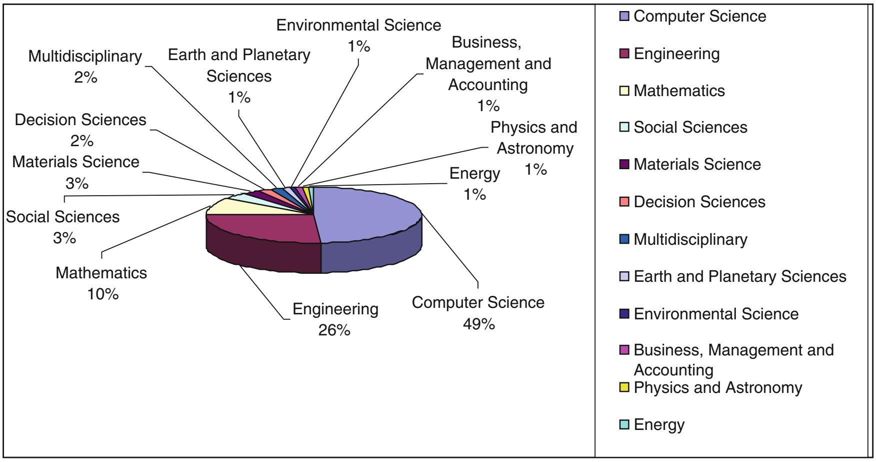


Fig. 6 Publication of neural network applied intrusion detection system development related articles in various domains

in a single graph. The figure depicts that Lecture Notes in Computer Science is the major journal publishing 25 articles on IDS based on neural network (11 %) followed by 24 articles in Computers and Security and 20 articles in Expert Systems with Applications (9 %) journals.

For more than 9 articles published in a domain are shown in the Fig. 6 to get information of different domains in which neural network based IDS articles are found. It is shown that computer science (49 %) is the major domain publishing 509

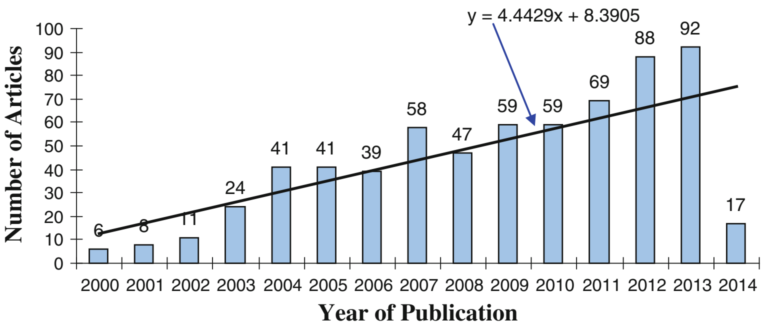


Fig. 7 Number of articles on neural network applied intrusion detection system development published during the year 2000–2014 and the research trend

articles followed by 273 articles in engineering (26 %) and 108 articles in mathematics (10 %).

In Fig. 7 the research trend based on the number of articles published between the years 2000–2013 has been shown to be increasing with R-squared value equals 0.9433 which is a good fit. The trend line in Fig. 7 for 2000–2014 is also increasing where the search on articles has been performed in February, 2014.

The next section has discussed the description of the data set applied in the development of the model for intrusion detection system.

3 KDD-99 Dataset

Mostly all the experiments on intrusion detection are done on KDDCUP'99 dataset, which is a subset of the 1998 DARPA Intrusion Detection Evaluation data set, and is processed, extracting 41 features from the raw data of DARPA 98 data set Stolfo et al. (2000) defined higher-level features that help in distinguishing between good normal connections from bad connections (attacks). This data can be used to test both host based and network based systems, and both signature and anomaly detection systems. A connection is a sequence of Transmission Control Protocol (TCP) packets starting and ending with well defined times, between which data flows from a source IP address to a target IP address under some well defined protocol. Each connection is labeled as normal, or as an attack, with exactly one specific attack type. Each connection record consists of about 100 bytes (https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html).

The data to be used in the model is organized and prepared to be used in the form of binary classification model. However the classification model needs to be evaluated based on certain metrics from their output results and discussed in the next section.

3.1 Evaluation Metrics

An elementary concern in the development of classification models is the evaluation of predictive accuracy (Guisan and Thuiller 2005; Barry and Elith 2006). The quantitative evaluation of the model is important as it helps in determining the ability of the model tom provide better solution for a specific problem and also assist in exploring the areas of model improvement. In the domain of binary predictions of anomaly and normal attacks, a confusion matrix (Table 1) known as contingency table or error matrix (Swets 1988) that represents the performance visualization of the predictive models of IDS that consists of two rows showing the actual class and two columns showing the predicted class. The aim is to check whether the system is confusing both classes. The IDSs are primarily distinguished binary classes: anomaly class (malicious, threats or abnormal data) and normal class (normal data points). Therefore, the proposed models generating normal-anomaly predictions of intrusion detection system are typically assessed in Table 1 through comparison of the predictions and developing a confusion matrix to predict the number of true positive (TP), false positive (FP), false negative (FN) and true negative (TN) cases. TP/TP+FN, is used as detection rate (DR) or sensitivity. It is also termed as recall in information retrieval Overall accuracy is a simple measure of accuracy that can be derived from the confusion matrix by calculating the proportion of correct prediction. Sensitivity is the proportion of observed normal attacks that are predicted as such, and therefore quantifies omission errors. Specificity is the proportion of observed anomaly attacks that are predicted as such, and therefore quantifies commission errors. Sensitivity and Specificity are independent of each other when compared across models. The most popular measure for the accuracy of yes–no predictions is Cohen's kappa (Shao and Halpin 1995; Segurado and Araujo 2004) which corrects the overall accuracy of model predictions by the expected random accuracy. The kappa statistic ranges from 0 to 1, where 1 indicates perfect agreement and values of zero indicate a performance no better than random (Cohen 1960). The principle benefits of kappa are for its simplicity and the reason that both commission and omission errors are accounted for in one parameter. In this paper we also introduced another measure known as the true skill statistic (TSS) for the performance of normal–anomaly classifier models, that still preserves the advantages of kappa.

In the next section a detail experiment and analysis demonstrated the efficacy of the proposed MLP in the development of IDS system based on the above discussed classification evaluation metrics.

Table 1 Confusion matrix

	Predicted class		
Actual class		Anomaly	Normal
	Anomaly	TP	FN
	Normal	FP	TN

4 Experiment and Analysis of Intrusion Detection System Based on MLP Algorithm

MLP is conceivably the most popular network architecture currently in use amongst the ANNs (Saftoiu et al. 2012). There are three layers of units: input layer, a hidden layer and an output layer in the architecture of MLP with feed-forward supervised learning. The proposed ANN architecture was implemented using the SPSS neural networks program using SPSS 16.0 (http://www-01.ibm.com/software/in/analytics/spss/downloads.html) in Windows XP environment. Neural Networks are nonlinear statistical data modeling approaches. ANNs can explore and extract nonlinear interactions among parameters to expose formerly unidentified associations among given input parameters and outcomes (Sall et al. 2007).

The Fig. 8 shows a feed forward architecture of the neural network because the connections in the network flow forward from the input layer to the output layer without any feedback loops. In this Fig. 8 the input layer contains the 39 predictors; one hidden layer contains unobservable nodes, or units. Based on some function of

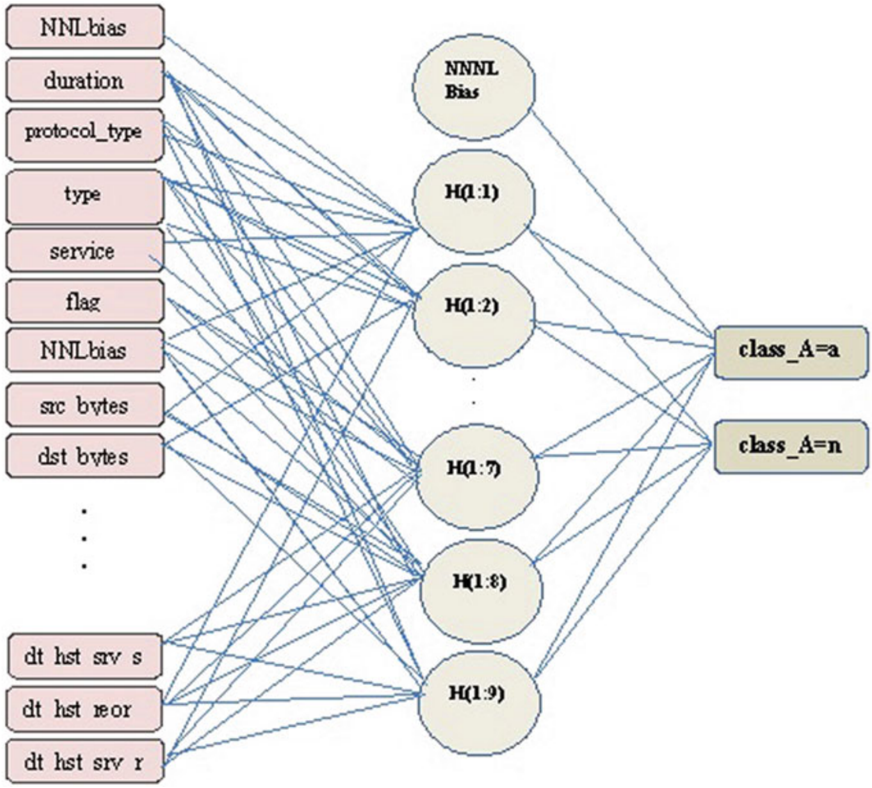


Fig. 8 Feedforward architecture with one hidden layer

Table 2 Case processing summary

		N	Percent (%)
Sample	Training	17,723	70.4
	Testing	7,468	29.6
Valid		25,191	100.0
Excluded		0	
Total		25,191	

the predictors represents the value of each hidden unit; that depends partly on the network type and on user-controllable condition. Anomaly and Normal from intrusion detection modeling point of view are being represented by the output layer as dependent variables. Since the class of response is a categorical variable with two classes, it is recoded as binary class variables. Each output node is some function of the hidden node that is also partly on the network type and on user-controllable condition. The proposed Multilayer Perceptron (MLP) model generates a predictive architecture for one dependent (target) variable to classify whether the attack class is anomaly or normal one.

In Table 2 the summary of case processing shows that 17,723 cases were assigned to the training sample and 7,468 to the testing sample.

Table 3 displays information on the neural network and is helpful for making sure that the specifications are accurate. The number of nodes in the input layer is 39 and similarly binary class out is represented by the two output units in output layer. The applied KDDCUP-99 dataset has 39 independent variables representing the input layer of the proposed model (duration, protocol_type, service, flag, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, num_failed_logins, logged_in, num_compromised, root_shell, su_attempted, num_root, num_file_creations, num_shells, num_access_files, is_guest_login, count, srv_count, serror_rate, srv_serror_rate, rerror_rate, srv_rerror_rate, same_srv_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_cnt, dt_hst_se_srv_rt, dt_host_diff_srv_rt, dt_hst_sm_src_prt_rt, dt_hst_srv_dif_ht_rt, dt_hst_seror_rt, dt_hst_srv_ser_rt,

Table 3 Network information

Input layer	Covariates number of units[a]	39 input variables from the KDDCUP 99 dataset
	Rescaling method for covariates	Standardized
Hidden layer(s)	Number of hidden layers	1
	Number of units in hidden layer 1[a]	9
	Activation function	Hyperbolic tangent
Output layer	Dependent variables	Class 1
	Number of units	2
	Activation function	Softmax
	Error function	Cross-entropy

[a] Excluding the bias unit

dt_hst_reor_rt, dt_hst_srv_rerr_rt). For the single default hidden layer there is 9 nodes. Rest of the information is default for the architecture.

Thus the developed IDS model required to be assessed based on performance measures using the model evaluation criteria discussed in the next section.

4.1 Measurement of Proposed Model Performance

In Table 4 the model summary shows information about the outcomes of training and applying the final network to the testing sample. Cross entropy error is shown since the output layer uses the softmax activation function using that the network tries to minimize the error during training. The confusion matrix provides the percentage of incorrect predictions. The execution of algorithm stopped when the maximum number of epochs reached and training has been completed ideally when the errors has converged.

In Table 5 the confusion matrix displays the useful outcomes of applying the network. For each case, the predicted response is anomaly if that cases's predicted pseudo-probability is greater than equal to 1 else it is normal attack. For each sample: Cells on the diagonal of the cross-classification of cases are correct predictions and off the diagonal of the cross-classification of cases are incorrect predictions.

Table 4 Model summary

Training	Cross entropy error	520.534
	Percent incorrect predictions	1.0 %
	Stopping rule used	1 consecutive step(s) with no decrease in error[a]
	Training time	00:00:11.969
Testing	Cross entropy error	331.762
	Percent incorrect predictions	1.3 %
Dependent variable: class		

[a] Error computations are based on the testing sample

Table 5 Confusion matrix

Sample	Observed	Predicted		
		a	n	Percent correct (%)
Training	a	8,142	73	99.1
	n	96	9,412	99.0
	Overall percent	46.5 %	53.5 %	99.0
Testing	a	3,484	43	98.8
	n	57	3,884	98.6
	Overall percent	47.4 %	52.6 %	98.7
Dependent variable: class				

Of the cases used to create the model, 9412 of the 9508 normal attacks are classified correctly (99 %) and 8142 of the 8215 anomaly attack types are classified correctly (99.1 %). Overall, 99.0 % of the training cases are classified correctly, corresponding to the 1 % incorrect shown in the Table 4 of model summary. Thus the model generates a better classification by correctly identifying a higher percentage of the cases. Classifications based upon the cases used to create the model tend to be too "optimistic" in the sense that their classification rate is inflated. The holdout sample facilitates to validate the model; here 98.8 % of these cases were correctly classified by the model. This suggests that, overall, the proposed model is in fact correct.

In Table 6 the model summary shows a couple of positive signs:

The percentage of incorrect predictions is roughly equal across training, testing, and holdout samples. The estimation algorithm stopped because the error did not decrease after a step in the algorithm. This further suggests that the original model did not over trained.

The confusion matrix in Table 7 shows that, the network does excellent at detecting anomaly than normal attacks. The detection rate and overall accuracy of

Table 6 Confusion matrix

Sample	Observed	Predicted		
		a	n	Percent correct (%)
Training	a	7,019	59	99.2
	n	70	7,981	99.1
	Overall percent	46.9 %	53.1 %	99.1
Testing	a	3,431	31	99.1
	n	53	4,044	98.7
	Overall percent	46.1 %	53.9 %	98.9
Holdout	a	1,190	12	99.0
	n	17	1,284	98.7
	Overall percent	48.2 %	51.8 %	98.8
Dependent variable: class				

Table 7 Model summary

Training	Cross entropy error	389.173
	% Incorrect predictions	0.9 %
	Stopping rule used	1 consecutive step(s) with no decrease in error[a]
	Training time	00:00:25.563
Testing	Cross entropy error	246.806
	Percent incorrect predictions	1.1 %
Holdout	Percent incorrect predictions	1.2 %
Dependent variable: class		

[a] Error computations are based on the testing sample

the testing outcomes have been calculated from Table 5 as 0.991045638, 0.988887419 respectively. Unfortunately, the single cutoff value (>zero) gives a very limited view of the predictive ability of the network, so it is not necessarily very useful for comparing competing networks rather focus should be on ROC curve

The Fig. 9 displays ROC curve that gives a visual display of the sensitivity and specificity for all possible cutoffs in a single plot, which is much cleaner and more powerful than a series of tables. The figure depicts here shows two curves, one for the category anomaly and one for the category normal. Since it is binary, the curves are symmetrical about a 45° line from the upper left corner of the chart to the lower right. This graph is based on the combination of training and testing samples.

The area under the curve is a numerical summary of the ROC curve, and the values in the table represent, for each category, the probability that the predicted pseudo-probability of being in that category is higher for a randomly chosen case in that category than for a randomly chosen case not in that category. In Table 8, for a randomly selected anomaly and randomly selected normal, there is a 0.999 probability that the model-predicted pseudo-probability of anomaly will be higher for the anomaly than for the normal. While the area under the curve is a useful one-statistic summary of the accuracy of the network, it is required to choose a specific criterion by which network intrusion is classified. The predicted-by-observed chart provides a visual start on this process (Fig. 10).

Fig. 9 ROC curve

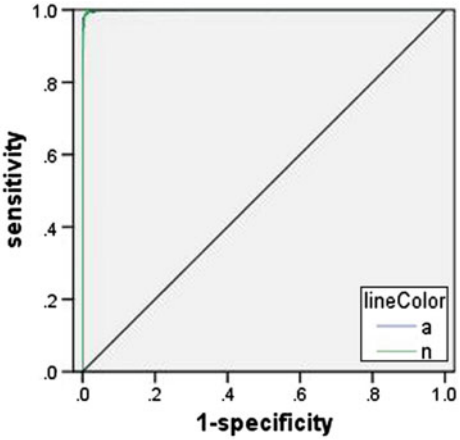
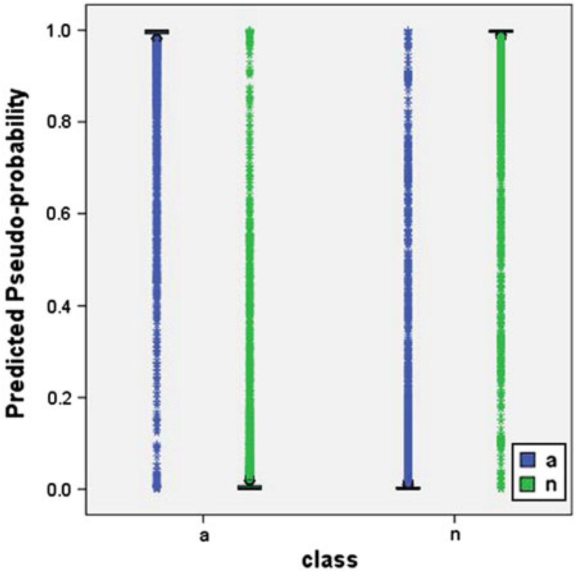


Table 8 Area under the curve

		Area
Class	a	0.999
	n	0.999

Fig. 10 Predicted-by-observed chart



For categorical dependent variables, the predicted-by-observed chart displays clustered boxplots of predicted pseudo-probabilities for the combined training and testing samples. The x axis corresponds to the observed response categories, and the legend corresponds to predicted categories.

The leftmost boxplot shows, for cases that have observed category anomaly, the predicted pseudo-probability of category anomaly. The portion of the boxplot above the 0.5 mark on the y axis represents correct predictions shown in the confusion matrix table. The portion below the 0.5 mark represents incorrect predictions. As shown in the confusion matrix table that the network is excellent at predicting cases with the anomaly category using the 0.5 cutoff, so only a portion of the lower whisker and some outlying cases are misclassified. The next boxplot to the right shows, for cases that have observed category anomaly, the predicted pseudo-probability of category normal. Since there are only two categories in the target variable, the first two boxplots are symmetrical about the horizontal line at 0.5.

The third boxplot shows, for cases that have observed category normal, the predicted pseudo-probability of category anomaly. It and the last boxplot are symmetrical about the horizontal line at 0.5. The last boxplot shows, for cases that have observed category normal, the predicted pseudo-probability of category normal. The portion of the boxplot above the 0.5 mark on the y axis represents correct predictions shown in the confusion matrix table. The portion below the 0.5 mark represents incorrect predictions. Remember from the confusion matrix table that the network predicts slightly more than half of the cases with the normal category using the 0.5 cutoff, so a good portion of the box is misclassified. Looking at the plot, it appears that by lowering the cutoff for classifying a case as normal from 0.5 to approximately 0.3—this is roughly the value where the top of the second box and the bottom of the fourth box are—that can increase the chance of correctly detecting possible intrusion without generating false alarm on normal attacks.

In the Fig. 11 cumulative gains chart demonstrates the percentage of the overall number of cases in a given category "gained" by targeting a percentage of the total number of cases. For example, the first point on the curve for the anomaly category is at (10, 20 %), meaning that if a dataset is scored with the network and sort all of the cases by predicted pseudo-probability of anomaly, it is expected that the top 10 % to contain approximately 20 % of all of the cases that actually take the category anomaly (attacks). Likewise, the top 20 % would contain approximately 45 % of the anomaly; the top 30 % of cases would contain 65 % of defaulters, and so on. If 100 % scored dataset is selected then all of the anomaly in the dataset will be obtained. The diagonal line is the "baseline" curve; if 10 % of scored dataset is selected at random, then it is expected to "gain" approximately 10 % of all of the cases that actually take the category anomaly. The farther above the baseline a curve lies, the greater the gain. The cumulative gains chart is used to help choose a classification cutoff by choosing a percentage that corresponds to a desirable gain, and then mapping that percentage to the appropriate cutoff value. What constitutes a "desirable" gain depends on the cost of Type I and Type II errors. That is, what is the cost of classifying a anomaly attack as a normal attack (Type I)? What is the cost of classifying a normal as a anomaly (Type II)? If any network parameter is the primary concern, then Type I error may be minimised; on the cumulative gains chart, this might correspond to generate alarm in the top 40 % of pseudo-predicted probability of anomaly, which captures nearly 90 % of the possible anomaly attacks

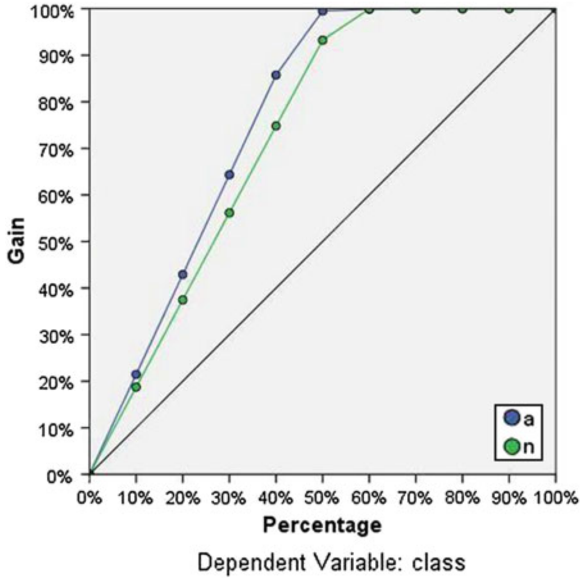


Fig. 11 Cumulative gains chart, dependent variable: class

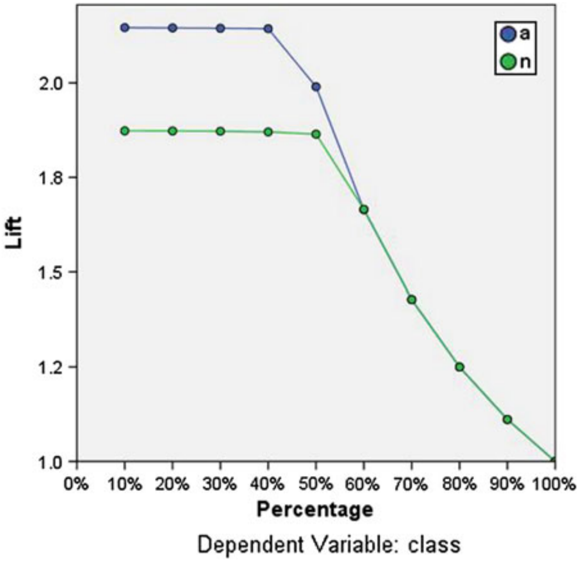


Fig. 12 Lift chart, dependent variable: class

Table 9 Comparison of three neural network architecture based on common measures of performance

	BPN	Recurrent	PCA
Sensitivity	0.9848	0.9650	0.9828
Specificity	0.9924	0.9306	0.9594
Overall accuracy	0.9889	0.9490	0.9719
Kappa statistic	0.9630	0.8970	0.9430
TSS	0.9772	0.8956	0.9422

but removes nearly half of total attacks. If a very large data set is the priority, then Type II error may be minimised. On the chart, this might correspond to rejecting the top 10 %, which captures 20 % of the anomaly and leaves most of KDD99 data set intact. Usually, both are major concerns, so a decision rule should have been chosen for classifying attacks that gives the best mix of sensitivity and specificity.

The lift chart in Fig. 12 is derived from the cumulative gains chart; the values on the y axis correspond to the ratio of the cumulative gain for each curve to the baseline. Thus, the lift at 10 % for the category Yes is 30 %/10 % = 3.0. It provides another way of looking at the information in the cumulative gains chart.

Note: The cumulative gains and lift charts are based on the combined training and testing samples (Table 9).

5 Conclusion

As described in the preceding section, MLP method has recognized them as a good choice for any existing intrusion detection system. This paper provides a state-of-the-art review of the applications of neural network to Intrusion Detection System. Following findings are significant in the research review of IDS:

- Artificial neural network based intrusion detection system development is an important research trend in intrusion detection domain.
- China has shown to be significantly contributing (32 %) followed by USA (16 %) and India (9 %) in terms of publication by affiliated country.
- The conference paper (65 %) has recognized as the major type of research documents followed by articles (30 %).
- Lecture Notes in Computer Science has emerged as leading journal that published 25 articles on IDS based on neural network (11 %) followed by 24 articles in Computers and Security and 20 articles in Expert Systems with Applications (9 %) journals.
- Undoubtedly the computer science (49 %) is shown to be the major domain publishing 509 articles followed by 273 articles in engineering (26 %) and 108 articles in mathematics (10 %).
- The current research trend based on the number of articles published between the years 2000–2013 has been shown to be increasing with R-squared value equals 0.9433 as a good fit. The trend line for 2000–2014 is also shown to be increased.

In this research, we have proposed architecture based on Multi Layer Perceptron neural network. The model builds the intrusion detection system learnt from the patterns of KDD99 data set. Based on the identified patterns, the architecture recognized attacks in the datasets using the back propagation neural network algorithm. The proposed neural network approach resulted with higher detection rate, a reduced amount of execution time. We continue our work in this direction in order to build an efficient intrusion detection model. When the proposed Back propagation neural network approach is compared with the other two approaches: Recurrent and PCA neural network based on the common measures of performance it is clearly visible as shown in Fig. 13 to outperform the performances of the other two approaches. Further work will be undertaken to increase the performance of the intrusion detection model and reduce the false alarm and efficiently handle the identification of correct anomaly dynamically.

Since the goal of this research was also to evaluate the performance of our proposed approach by comparing with other six approaches available in literature in terms of three measures of performance: detection rate, accuracy rate and computation time of the intrusion detection (Table 10). The comparative research findings from Table 10 has revealed that the proposed approach has succeeded in achieving increased rate of anomaly detection, reduced false alarm and at the same time minimal execution time for the development of intrusion detection system. In KPCA and SVM approach, the accuracy rate is 99.2 % (98.89 % accuracy in case of

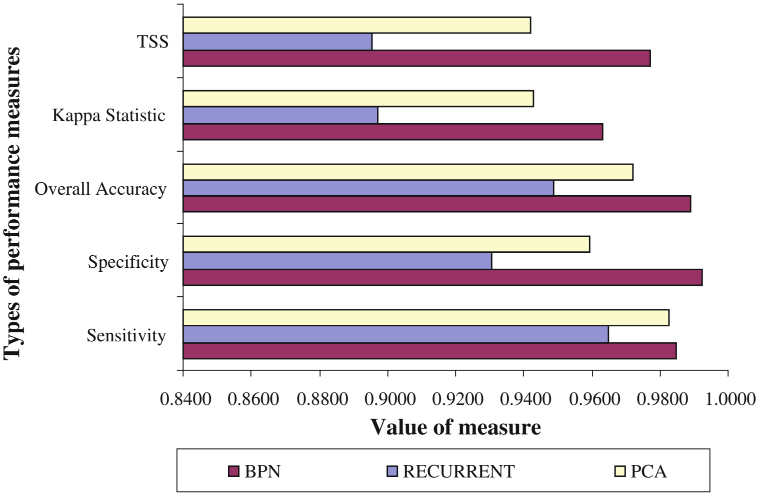


Fig. 13 Comparison of the measure of performance for three neural network architecture applied to develop intrusion detection system

Table 10 Comparative performance of literature available approaches used with proposed multilayer perceptron approach based on detection rate, accuracy and computation time

Approach used	References	Detection rate testing %	Accuracy testing %	Computation time (s)	Dataset used in experiment
KPCA and SVM	Kuanf et al. (2012)	–	99.2 (training: 99.975)	407.918466	KDD dataset 6000 sample-4000 for training, 2000 for testing (Han 2012)
Resilient back propagation neural network	Naoum et al. (2012)	94.7	–	–	KDD dataset (Naoum et al. 2005)
Decision tree based light weight intrusion detection using wrapper approach	Sivatha Sindhu et al. (2012)	98.38	–	–	KDD dataset (Sivatha Sindhu et al. 2012)
Neural network	Devaraju and Ramakrishnan (2011)	–	97.5	–	KDD dataset (Kuanf et al. 2012)
BPNN	Mukhopadhyay et al. (2011)	–	–	–	KDD dataset (Mukhopadhyay et al. 2011)
SOM	Ibrahim et al. (2013)	92.37		–	KDD 99
Our proposed approach[a]	–	99.10	98.89	11.969	KDD 20 % dataset

[a] The data is taken from 70 to 30 dataset as it is giving better detection rate

proposed approach) however detection rate is unknown with a larger computation time. In the rest of the result of Table 10 the detection rate and computation time of the proposed MLP approach are superior.

The future work should be directed towards developing hybrid neural network to increase the efficiency of intrusion detection and to deal the dynamic large data stream to secure from network intrusion.

References

Anderson, J. (1995). *An introduction to neural networks*. Cambridge: MIT Press.

Anyanwu, L. O., Keengwe, J., & Arome, G. A. (2011). Scalable intrusion detection with recurrent neural networks. *International Journal of Multimedia and Ubiquitous Engineering, 6*(1), 21–28.

Aziz, A. S. A., Azar, A. T., Hassanien, A. E., & Hanafy, S. E. O. (2012). Continuous features discretization for anomaly intrusion detectors generation. In *The 17th Online World Conference on Soft Computing in Industrial Applications* (*WSC17*), December 10–21.

Aziz, A. S. A., Azar, A. T., Hassanien, A. E., & Hanafy, S. E. O. (2014). Continuous features discretization for anomaly intrusion detectors generation. In *Soft computing in industrial applications* (pp. 209–221). Switzerland: Springer International Publishing.

Abdel-Aziz, A. S., Hassanien, A. E., Azar, A. T., & Hanafi, S. E. O. (2013). Machine learning techniques for anomalies detection and classification. *Advances in security of information and communication networks* (pp. 219–229). Berlin Heidelberg: Springer.

Barry, S., & Elith, J. (2006). Error and uncertainty in habitat models. *Journal of Applied Ecology, 43*(3), 413–423.

Behjat, A. R., Vatankhah, N., & Mustapha, A. (2014). Feature subset selection using genetic algorithm for intrusion detection system. *Advanced Science Letters, 20*(1), 235–238.

Bezdek, J. C. (1994). *What is computational intelligence? Computational intelligence imitating life* (pp. 1–12). New York: IEEE Press.

Chebrolu, S., Abraham, A., & Thomas, J. P. (2005). Feature deduction and ensemble design of intrusion detection systems. *Computers and Security, 24*(4), 295–307.

Chittur, A. (2001). Model generation for an intrusion detection system using genetic algorithms. High School Honors Thesis, Ossining High School. In Cooperation with Columbia Univ. Accessed on November 27, 2013.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46.

Dębska, B., & Guzowska-Świder, B. (2011). Application of artificial neural network in food classification. *Analytica Chimica Acta, 705*(1), 283–291.

Denning, D. E. (1987). An intrusion-detection model. *IEEE Transactions on Software Engineering, 13*(2), 222–232.

Devaraju, S., & Ramakrishnan, S. (2011). Performance analysis of intrusion detection system using various neural network classifiers. In Recent Trends in Information Technology (ICRTIT), June 2011 International Conference on (pp. 1033–1038). IEEE.

Eid, H. F., Azar, A. T., & Hassanien, A. E. (2013, January). Improved real-time discretize network intrusion detection system. In *Proceedings of seventh international conference on bio-inspired computing: theories and applications* (*BIC-TA 2012*) (pp. 99–109). India: Springer.

El Kadhi, N., Hadjar, K., & El Zant, N. (2012). A mobile agents and artificial neural networks for intrusion detection. *Journal of Software, 7*(1), 156–160.

Eskin, E., Arnold, A., Prerau, M., Portnoy, L., & Stolfo, S. (2002). A geometric framework for unsupervised anomaly detection. *Applications of data mining in computer security* (pp. 77–101). US: Springer.

Faysel, M. A., & Haque, S. S. (2010). Towards cyber defense: research in intrusion detection and intrusion prevention systems. *IJCSNS International Journal of Computer Science and Network Security, 10*(7), 316–325.

Feizollah, A., Anuar, N. B., Salleh, R., Amalina, F., Ma'arof, R. U. R., & Shamshirband, S. (2014). A study of machine learning classifiers for anomaly-based mobile Botnet detection. *Malaysian Journal of Computer Science, 26*(4), 251–265.

Gong, R. H., Zulkernine, M., & Abolmaesumi, P. (2005, May). A software implementation of a genetic algorithm based approach to network intrusion detection. In *Sixth international conference on software engineering, artificial intelligence, networking and parallel/distributed computing*, *2005 and first ACIS international workshop on self-assembling wireless networks* (*SNPD/SAWN 2005*) (pp. 246–253). IEEE.

Guisan, A., & Thuiller, W. (2005). Predicting species distribution: Offering more than simple habitat models. *Ecology Letters, 8*(9), 993–1009.

Gupta, B. B., Joshi, R. C., & Misra, M. (2012). ANN based scheme to predict number of Zombies in a DDoS attack. *IJ Network Security, 14*(2), 61–70.

Han, L. (2012). Research of K-MEANS algorithm based on information Entropy in Anomaly Detection. In Multimedia Information Networking and Security (MINES), November 2012 Fourth International Conference on (pp. 71-74). IEEE.

Haykin, S. (2005). *Neural networks a comprehensive foundation*. New Delhi: Pearson Education.

Heady R., Luger G., Maccabe A., & Servilla M. (1990, August). The architecture of a network level intrusion detection system. Technical report, Computer Science Department, University of New Mexico.

Hwang, R. C., Chen, Y. J., & Huang, H. C. (2010). Artificial intelligent analyzer for mechanical properties of rolled steel bar by using neural networks. *Expert Systems with Applications, 37* (4), 3136–3139.

Ibrahim, L. M., Basheer, D. T., & Mahmod, M. S. (2013). A comparison study for intrusion database (Kdd99, Nsl-Kdd) based on self organization map (SOM) artificial neural network. *Journal of Engineering Science and Technology, 8*(1), 107–119.

Khashei, M., Rezvan, M. T., Hamadani, A. Z., & Bijari, M. (2013). A bi-level neural-based fuzzy classification approach for credit scoring problems. *Complexity, 18*(6), 46–57.

Kuanf, F., Xu, W., Zhang, S., Wang,Y., & Liu, K. (2012). A novel Approach of KPCA and SVM for Intrusion Detection, *Journal of Computational Information Systems*, pp 3237–3244.

Kuo, R. J., Wang, Y. C., & Tien, F. C. (2010). Integration of artificial neural network and MADA methods for green supplier selection. *Journal of Cleaner Production, 18*(12), 1161–1170.

Laskov, P., Düssel, P., Schäfer, C., & Rieck, K. (2005). Learning intrusion detection: Supervised or unsupervised? In *Image analysis and processing—ICIAP 2005* (pp. 50–57). Berlin Heidelberg: Springer.

Lee, W., Stolfo, S. J., & Mok, K. W. (1999). A data mining framework for building intrusion detection models. In *Proceedings of the 1999 IEEE symposium on security and privacy* (pp. 120–132). IEEE.

Liao, Y., & Vemuri, V. R. (2002). Use of K-nearest neighbor classifier for intrusion detection. *Computers and Security, 21*(5), 439–448.

Liu, J. (2013). An adaptive intrusion detection model based on ART2 neural network. *Journal of Computational Information Systems, 9*(19), 7775–7782.

Louvieris, P., Clewley, N., & Liu, X. (2013). Effects-based feature identification for network intrusion detection. *Neurocomputing, 121*, 265–273.

McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (1955). A proposal for the dartmouth summer research project on artificial intelligence, August 31, 1955. *AI Magazine, 27*(4), 12.

McCarthy, J. (2007). What is artificial intelligence. url: http://www-formal.stanford.edu/jmc/whatisai.html. (accessed on 22 November 2013)

Mukhopadhyay, I., Chakraborty, M., Chakrabarti, S., & Chatterjee, T. (2011). Back propagation neural network approach to Intrusion Detection System. In *Recent Trends in Information Systems (ReTIS), December 2011 International Conference* on (pp. 303–308). IEEE.

Naoum, R. S., Abid, N. A., Al-Sultani, Z. N. (2005) "An enhanced Resilient backpropagation artificial neural network for Intrusion detection", International Journal of Computer Science and Network Security, 2005, *12*(3), 11–16.

Pan Z., Chen, S., Hu, G., & Zhang, D. (2003). Hybrid neural network and C4.5 for misuse detection. In *Proceedings of the second international conference on machine learning and cybernetics* (Vol. 4, pp. 2463–2467). IEEE.

Peláez, J. I., Doña, J. M., Fornari, J. F., & Serra, G. (2014). Ischemia classification via ECG using MLP neural networks. *International Journal of Computational Intelligence Systems, 7*(2), 344–352.

Peng, Y., Wang, Y., Niu, Y., & Hu, Q. (2014). Application study on intrusion detection system using IRBF. *Journal of Software, 9*(1), 177–183.

Saftoiu, A., Vilmann, P., Gorunescu, F., Janssen, J., Hocke, M., & Larsen, M., et al. (2012). Efficacy of an artificial neural network-based approach to endoscopic ultrasound elastography in diagnosis of focal pancreatic masses. *Clinical Gastroenterology Hepatology, 10*(1), 84–90.

Sall, J., Creighton, L., & Lehman, A. (2007). Safari tech books online. JMP start statistics a guide to statistics and data analysis using JMP. *SAS press series* (4th edn.). Cary, N.C.: SAS Pub.

Segurado, P., & Araujo, M. B. (2004). An evaluation of methods for modelling species distributions. *Journal of Biogeography, 31*(10), 1555–1568.

Shao, G., & Halpin, P. N. (1995). Climatic controls of eastern North American coastal tree and shrub distributions. *Journal of Biogeography*, 1083–1089.

Sheikhan, M., & Sharifi Rad, M. (2011). Intrusion detection improvement using GA-optimized fuzzy grids-based rule mining feature selector and fuzzy ARTMAP neural network. *World Applied Sciences Journal, 14*, 772–781.

Sheikhan, M., & Sharifi, Rad M. (2013). Using particle swarm optimization in fuzzy association rules-based feature selection and fuzzy ARTMAP-based attack recognition. *Security and Communication Networks, 6*(7), 797–811.

Sivatha Sindhu, S. S., Geetha, S., & Kannan, A. (2012). Decision tree based light weight intrusion detection using a wrapper approach. *Expert Systems with applications*, *39*(1), 129–141.

Stolfo, S. J., Fan, W., Lee, W., Prodromidis, A., & Chan, P. K. (2000). Cost-based modeling for fraud and intrusion detection: Results from the JAM project. In *Proceedings of the DARPA information survivability conference and exposition, 2000* (*DISCEX'00*) (Vol. 2, pp. 130–144). IEEE.

Swets, J. A. (1988). Measuring the accuracy of diagnostic systems. *Science, 240*(4857), 1285–1293.

Tiwari, P. (2002). Intrusion detection. Technical Report, Department of Electrical Engineering, Indian Institute of Technology, Delhi.

Tuncer, T., & Tatar, Y. (2012). Implementation of the FPGA based programmable embedded intrusion detection system. *Journal of the Faculty of Engineering and Architecture of Gazi University, 27*(1), 59–69.

Valero, S., Senabre, C., López, M., Aparicio, J., Gabaldon, A., & Ortiz, M. (2012). Comparison of electric load forecasting between using SOM and MLP neural network. *Journal of Energy and Power Engineering, 6*(3), 411–417.

Wang, G., Hao, J., Ma, J., & Huang, L. (2010). A new approach to intrusion detection using artificial neural networks and fuzzy clustering. *Expert Systems with Applications, 37*(9), 6225–6232.

Wang, J. H., Liao, Y. L., Tsai, T. M., & Hung, G. (2006). Technology-based financial frauds in Taiwan: Issues and approaches. In *SMC* (pp. 1120–1124).

Wu, S. X., & Banzhaf, W. (2010). The use of computational intelligence in intrusion detection systems: A review. *Applied Soft Computing, 10*(1), 1–35.

Xiang, Z., Zhu, J., Han, W., & Ding, J. (2013). On the capability of SOINN based intrusion detection systems. *Journal of Computational Information Systems, 9*(3), 941–949.

Yang, S., Yang, Y., Shen, Q., & Huang, H. (2013). A method of intrusion detection based on semi-supervised GHSOM. In *Jisuanji Yanjiu yu Fazhan/Computer Research and Development. Jisuanji Yanjiu yu Fazhan/Computer Research and Development*, *November 2013* (Vol. 50(11), pp. 2375–2382).

Yao, J. T., Zhao, S. L., & Saxton, L. V. (2005). A study on fuzzy intrusion detection. In B. V. Dasarathy (Ed.), In *Proceedings of SPIE vol. 5812*, *data mining, intrusion detection, information assurance*, *and data networks security*, 28 March–1 April 2005 (pp. 23–30). Orlando, Florida, USA, Bellingham, WA: SPIE.

Zainaddin, A., Asyiqin, D., & Mohd Hanapi, Z. (2013). Hybrid of fuzzy clustering neural network over NSL dataset for intrusion detection system. *Journal of Computer Science, 9*(3), 391–403.

Zhao, Y., Zha, Y., & Zha, X. (2013). Network intrusion detection based on IPSO-BPNN. *Information Technology Journal, 12*(14), 2719–2725.