Ganapathy *et al. EURASIP Journal on Wireless Communications and Networking* 2013, **2013**:271
http://jwcn.eurasipjournals.com/content/2013/1/271

EURASIP Journal on
Wireless Communications and Networking
a SpringerOpen Journal

REVIEW Open Access

Intelligent feature selection and classification techniques for intrusion detection in networks: a survey

Sannasi Ganapathy[*], Kanagasabai Kulothungan, Sannasy Muthurajkumar, Muthusamy Vijayalakshmi, Palanichamy Yogesh and Arputharaj Kannan

Abstract

Rapid growth in the Internet usage and diverse military applications have led researchers to think of intelligent systems that can assist the users and applications in getting the services by delivering required quality of service in networks. Some kinds of intelligent techniques are appropriate for providing security in communication pertaining to distributed environments such as mobile computing, e-commerce, telecommunication, and network management. In this paper, a survey on intelligent techniques for feature selection and classification for intrusion detection in networks based on intelligent software agents, neural networks, genetic algorithms, neuro-genetic algorithms, fuzzy techniques, rough sets, and particle swarm intelligence has been proposed. These techniques have been useful for effectively identifying and preventing network intrusions in order to provide security to the Internet and to enhance the quality of service. In addition to the survey on existing intelligent techniques for intrusion detection systems, two new algorithms namely intelligent rule-based attribute selection algorithm for effective feature selection and intelligent rule-based enhanced multiclass support vector machine have been proposed in this paper.

Keywords: Survey; Intrusion detection system; Neural networks; Fuzzy systems; Swarm intelligence; Particle swarm intelligence

Review

Intrusion detection systems

Recently, Internet has become a part and parcel of daily life. The current internet-based information processing systems are prone to different kinds of threats which lead to various types of damages resulting in significant losses. Therefore, the importance of information security is evolving quickly. The most basic goal of information security is to develop defensive information systems which are secure from unauthorized access, use, disclosure, disruption, modification, or destruction. Moreover, information security minimizes the risks related to the three main security goals namely confidentiality, integrity, and availability.

Various systems have been designed in the past to identify and block the Internet-based attacks. The most important systems among them are intrusion detection systems (IDS) since they resist external attacks effectively. Moreover, IDSs provide a wall of defense which overcomes the attack of computer systems on the Internet. IDS could be used to detect different types of attacks on network communications and computer system usage where the traditional firewall cannot perform well. Intrusion detection is based on an assumption that the behavior of intruders differ from a legal user [1]. Generally, IDSs are broadly classified into two categories namely anomaly and misuse detection systems based on their detection approaches [2,3]. Anomaly intrusion detection determines whether deviation from the established normal usage patterns can be flagged as intrusions. On the other hand, misuse detection systems detect the violations of permissions effectively. Intrusion detection systems can be built by using intelligent agents and classification techniques. Most IDSs work in two phases namely preprocessing phase and intrusion detection phase. The intrusions identified by the IDSs can be prevented effectively by developing an intrusion

[*] Correspondence: ganapathy.sannasi@gmail.com
Department of Information Science and Technology, College of Engineering Guindy, Anna University, Chennai 25, Tamilnadu, India

prevention system. This paper mainly provides a survey on intelligent techniques proposed for developing IDSs. In addition, it explains about a new IDS which has been developed using two proposed algorithms namely intelligent rule-based attribute selection algorithm and intelligent rule-based enhanced multiclass support vector machine (IREMSVM).

Intelligent intrusion detection systems

Intelligent IDSs are the ones considered to be intelligent computer programs situated in either a host or a network which analyzes the environment and acts flexibly to achieve higher detection accuracy [4,5]. These programs compute the actions to be performed on the environment both by learning the environment and by firing rules of inference [6]. Intelligent IDSs are capable of decision making and constraint checking. In most intelligent systems, either rules are fired or agents are used for decision making. Moreover, a set of static agents or a set of mobile and static agents have been used to achieve a single goal. Intelligent intrusion detection systems have been developed by proposing intelligent techniques for preprocessing and effective classification. Such IDSs have provided better detection rate in comparison with the other approaches.

Intelligent preprocessing techniques

Feature selection (or preprocessing) consists of detecting the relevant features and discarding the irrelevant ones, with the goal of obtaining a subset of features that describe the given problem properly with a minimum degradation of performance. It has several advantages [7], such as improving the performance of the machine learning algorithms, data understanding, gaining knowledge about the process and helping to visualize it, data reduction, limiting storage requirements, and helping in reducing processing costs.

There are two main models that deal with feature selection: filter methods and wrapper methods [8]. While wrapper models involve optimizing a predictor as part of the selection process, filter models rely on the general characteristics of the training data to select features with independence of any predictor. Wrapper models tend to give better results and this model is more precise than the filter model.

Intelligent classification techniques

Classification [9,10] is used to learn a model called classifier from a set of labeled data instances called training and then to classify a test instance into one of the classes using the learned model known as testing. Classification-based anomaly detection techniques operate in a similar two-phase fashion. The training phase learns a classifier using the available labeled training data. The testing phase classifies a test instance as normal or anomalous, using the classifier. Classification-based anomaly detection techniques operate under either one-class classifier or multi-class classifier.

One-class-classification-based anomaly detection techniques assume that all training instances have only one class label. Such techniques learn a discriminative boundary around the normal instances using a *one-class classification algorithm.* Any test instance that does not fall within the learned boundary is declared as anomalous.

Multi-class-classification-based anomaly detection techniques assume that the training data contains labeled instances belonging to multiple normal classes [11]. Such anomaly detection techniques teach a classifier to distinguish between each normal class and the rest of the classes. A test instance is considered anomalous if it is not classified as normal by any of the classifiers. Some techniques in this subcategory associate a confidence score with the prediction made by the classifier. If none of the classifiers are confident in classifying the test instance as normal, the instance is declared to be anomalous.

Many intelligent classification techniques namely decision trees, neural networks (NN), navie bayes and fuzzy set-based approach are available in the literature. This paper considers the most important intelligent classification techniques for comparison.

Decision trees

A decision tree [12] is a tree where each non-terminal node represents a test or decision on the considered data item. Choice of a certain branch depends upon the outcome of the test. To classify a particular data item, the decision tree algorithms start at the root node and follow the assertions down until it reaches a terminal node (or leaf). A decision is made when a terminal node is approached. Decision trees can also be interpreted as a special form of a rule set, characterized by their hierarchical organization of rules.

Neural networks

NN [13] are those systems modeled based on the human brain working. As the human brain consists of millions of neurons that are interconnected by synapses, a neural network is a set of connected input or output units in which each connection has a weight associated with it. The network learns in the learning phase by adjusting the weights so as to be able to predict the correct class label of the input. An artificial neural network consists of connected set of processing units. The connections have weights that determine how one unit will affect another. Subsets of such units act as input and output nodes, and the remaining nodes constitute the hidden layer. By assigning activation to each of the input node and allowing them to propagate through the hidden layer nodes to the output nodes, neural network performs a functional mapping from input values to output values.

Naive Bayesian classifiers

Naive Bayesian classifiers [14] use Baye's theorem to classify the new instances of a data sample X. Each instance is a set of attribute values described by a vector, $X = (x_1, x_2,\ldots, x_n)$. Considering m classes, the sample X is assigned to the class C_i if and only if $P(X \mid C_i)\ P(C_i) > P(X \mid C_j)\ P(C_j)$ for all i and j in (1, m) such that j < > i. The sample belongs to the class with maximum posterior probability for the sample. For categorical data, $P(X_k \mid C_i)$ is calculated as the ratio of frequency of value X_k for attribute A_k and the total number of samples in the training set. For continuous valued attributes, Gaussian distribution can be assumed without loss of generality. In naive Bayesian approach, the attributes are assumed to be conditionally independent. In spite of this assumption, naive Bayesian classifiers give satisfactory results because focus is on identifying the classes for the instances, not the exact probabilities. Applications like spam mail classification and text classification can use naïve Bayesian classifiers. Theoretically, Bayesian classifiers are least prone to errors. The limitation is the requirement of the prior probabilities. The amount of probability information required is exponential in terms of number of attributes, number of classes, and the maximum cardinality of attributes. With increase in number of classes or attributes, the space and computational complexity of Bayesian classifiers increase exponentially.

Fuzzy sets

Fuzzy sets [15,16] form a key methodology for representing and processing uncertain information. Uncertainty arises in many forms in today's databases: imprecision, non-specificity, inconsistency, vagueness, etc. Fuzzy sets exploit uncertainty in an attempt to make system complexity manageable. As such, fuzzy sets constitute a powerful approach not only to deal with incomplete, noisy, or imprecise data but also to help in developing uncertain models of the data that provide smarter and smoother performance than traditional systems.

Dataset and performance metrics

Dataset

Since 1999, KDD’99 [17,18] has been the most widely used data set for the evaluation of anomaly intrusion detection methods. The KDD’99 Cup data set was prepared by Stolfo et al. [19] and was built based on the data captured in DARPA’98 IDS evaluation program [20,21]. This dataset was taken from the Third International Knowledge Discovery and Data Mining Tools Competition (KDD Cup 99). In this data set, each connection record is described by 41 attributes. The list of attributes consists of both continuous-type and discrete type variables, with statistical distributions varying drastically from each other, which makes the intrusion detection a very challenging task. The simulated attacks fall in one of the following four categories namely, denial of service (DoS), user to root (U2R), remote to local (R2L), and probe attacks. KDD’99 features are classified into three groups namely, basic features, traffic features, and content features. Traffic features are also classified into two types namely same host features and same service features.

Features present in KDD’99 Cup data set are grouped into three categories and are discussed below.

a. Basic Features: Basic features comprises of all the attributes that are extracted from a TCP/IP connection. These features are extracted from the packet header and includes src_bytes, dst_bytes, protocol etc.
b. Content Features: These features are used to evaluate the payload of the original TCP packet and looks for suspicious behavior in the payload portion. This includes features such as the number of failed login attempts, number of file creation operations etc. Moreover, most of the R2L and U2R attacks don’t have any frequent sequential patterns. This is due to the fact that DoS and Probing attacks involve many connections to some host(s) in a very short duration of time but the R2L and U2R attacks are embedded in the data portions of the packets, and generally involves only a single connection. Hence, content based features are used to detect the attacks.
c. Traffic Features: These include features that are computed with respect to a window interval and are divided into two categories
   i) “Same host” features: These features are derived only by examining the connections in the past 2 seconds that have the same destination host as the current connection, and compute statistics related to protocol behavior, service etc.
   ii) “Same service” features: These features examine only the connections in the past 2 seconds that have the same service as the current connection. The above two types are called “time based traffic features”.

Apart from these, there are various slow probing attacks that scan the hosts or ports using time interval greater than 2 seconds. As a result, these types of attacks do not generate intrusion patterns with a time window of 2 seconds. To overcome this problem, the “same host” and “same service” features are normally re-computed using a connection window of 100 connections.

Performance metrics

By using its ability to make correct predictions, the effectiveness of the IDS is evaluated based on four possible metrics namely true negative rate (TNR), true positive rate (TPR), false positive rate (FPR), and false negative rate

(FNR). If the actual class in the validation dataset is normal and is classified as normal, then TPR is incremented by 1 for each of the record. TNR is obtained if abnormal records are classified as abnormal records. FNR is obtained if normal record is classified as an anomaly record, and FPR is attained if abnormal record is classified as normal. Moreover, the most popular performance metrics for IDSs namely the detection rate (DR) and the false positive rate (FPR) are considered in this paper for effective analysis. DR is a ratio between number of anomaly (normal) correctly classified and total number of anomaly (normal). FPR is a ratio between number of anomalies incorrectly classified and total number of anomalies. An IDS should have a high DR and a low FPR. Other commonly used combinations include precision and recall, or sensitivity and specificity.

Testing scenario

In this paper, the KDD cup data have been used for evaluating the most prominent algorithms available in the literature for feature selection and classification. A subset that consists of 10% of the records available from the KDD cup data set were used to evaluate the algorithms discussed in this paper due to the large number of records present in the data set. The data set were chosen in such a way this subset reflects all the properties necessary for distributing the four types of attacks and also the normal records. Moreover, tenfold cross validation was carried out on this subset of data by dividing this chosen data set into ten parts in which each tie nine parts are used for training and one part for testing. The experiments were carried out using Java programs written and tested for different algorithms presented in this paper. The results obtained are discussed in this paper based on the data analysis using KDD cup data set.

Works on intelligent IDSs

In recent times, a lot of computational intelligence approaches were used for effective intrusion detection. The techniques include intelligent agent-based system, neural network-based IDSs, genetic algorithms, fuzzy and rough sets, particle swarm intelligence, and soft computing techniques.

Survey of intelligent agent-based systems

Intelligent agent-based systems are classified into four types namely simple agents, multi agents, mobile agents, and ant-based agents.

Static agents

In the past, two types of static agents have been proposed namely simple and multi-agents. Simple agents have the capability to sense the environment and to act upon them. Bakar et al. [22] proposed a new agent-based approach for intrusion detection using rough set-based classification technique that uses simple agents. This technique generates rules from the data available on a large database and has mechanisms through rough sets to handle noise and uncertainty in data. However, provision of a rough classification model or rough classifier is computationally expensive, especially in its reduced computation phase.

Multi agent systems are employed to attain inherently robust solutions to many robotic applications like exploration, surveillance, patrolling, target tracking, and intelligent transportation. In these situations, agents could perform different and possibly independent tasks, but at the same time, they cooperate in order to guarantee the entire system's safety. Cooperation among agents is obtained through a shared set of rules according to which all agents are supposed to plan their actions. Adriano Fagiolini et al. [23] addressed the major problem where the uncooperative behavior in a team of hybrid agents are detected and proposed the architecture of a decentralized monitor to be embedded on the agents. By the path of this monitoring process, each agent was able to establish whether its neighbors are cooperative or not. The major advantages of this agent architecture are the scalability and decentralization. The disadvantage of this agent is not considering the implementation aspects of such monitors. Xiaodong Zhu et al. [24] presented a multi-agent-based intrusion detection system named multi-agent-based intelligent intrusion detection system. The learning agent module in that system is self-adjusting and learns the network-based audit data and the host-based audit data, with a capability of learning more than one technique of data mining, such as association rules and so on. The learning agent also could produce rules, and the detection agent could detect audit data according to these rules and respond to them. The experimental results show that their system has very high self-adapting ability, intelligence, and expansibility.

In the work of Gou Xiantai et al. [25], they have focused on the first-class automatic reaction of containment since it is quite practical and easy to set up a worm containment system for a metropolitan area networks (MAN) but not for the whole Internet. Hence,multi-agent system for worm detection and containment in MAN was given to limit the propagation of worms in MAN. The major advantage of the system are that it could prevent the whole MAN from being fallen down because of the worm scan and the worm attack such as distributed denial of service (DDoS), and it is very effective in blocking random scanning worms that most commonly have been encountered. The disadvantage of this system is that it is not appropriate for restricting other types of worms such as the flash worms, topological worms [26] and random scanning worms that infect networks faster than any other type of worms. Due to various drawbacks, they tend to be very noisy and hence waste a lot of network bandwidth and crash the routers. So, it is very important to have some automatic reaction mechanism to limit the propagation of such type of worms.

Mobile agents

Mobile agents move from one host to another to carry out specific tasks. Ghenima Bourkache et al. [27] proposed a prototype of architecture of an anomaly distributed intrusion detection system for ad hoc networks that functions using a society of mobile and reactive agents carrying out intelligent and distributed intrusion detection. Their distributed model aimed at solving the problems faced by the hierarchical intrusion detection systems namely lower detection workload and great overhead. It proposes a technique for finding the main cause of the attack by the response engine in order to isolate the intruder from the network. Another work that uses mobile agents for intrusion detection was proposed by Wang et al. [28].

An integrated framework was proposed in [29] to guide the design of a mobile agent-based network management system namely the mobile agent-based framework for security-enhanced autonomous network and system management. This framework has offered two distinct advantages: (1) the provision of a secure agent-based management infrastructure and (2) the capability of achieving enhanced network management functionalities. They proposed two novel security schemes, namely the visibility protection scheme and the visa-based authentication scheme, for protection of management information and authentication and resource access control of management agents, respectively. Mobile agents could facilitate the implementation of robust, attack-resistant IDS architectures [30]. Agents were supposed to relocate when sensing danger or suspicious activity, clone for redundancy or replacement, operate autonomously and asynchronously from where created, collaborate and share knowledge, and be self organizing. Moreover, agents are amenable to genetic diversity, which also helped to avoid attacks aimed at circumventing the known and stable detection mechanisms of IDS.

Ant based agents

The ant colony optimization (ACO) algorithm depicted a probabilistic technique for solving computational problems which could be reduced to find good paths through graphs based on the strategies of real ants [31]. It was initially proposed in 1992 by Colorni, Dorigo, and Maniezzo [29,30]. In ACO, each artificial ant is considered as a simple agent, communicating with other ants only indirectly and by affecting changes to a common environment.

Chi-Ho Tsang et al. [32] presented a multi-agent IDS architecture for scalable intrusion detection and prevention in large switched networks in industrial plants. This unsupervised anomaly detection model based on ant colony was proposed and implemented in the decision agents in order to search heuristically for the near-optimal clustering with compact structure. The empirical results proved that this system could significantly improve the overall performance of existing ant-based clustering algorithms. On the other hand, this ant colony-based model could automatically determine the number of clusters that was critically required to be input in other clustering algorithms such as k-means, fuzzy c-means, and e-m clustering. They proved that ant agents are useful for reducing false positives in IDS.

Neural networks based IDSs

Neural networks are composed of basic units somewhat analogous to neurons. These units are linked with each other using the connection whose strength is modifiable as a result of a learning process or algorithm. Each of these units were integrated independently (in parallel) the information provided by its synapses in order to evaluate its state of activation. The unit response was then a linear or nonlinear function of its activation. Linear algebra concepts are used, in general, to analyze linear units, with eigenvectors and eigenvalues being the core concepts involved. This analysis made clear about the strong similarity between linear neural networks and the general linear model developed by statisticians. The linear models presented here are the perceptron and the linear associator. The behavior of nonlinear networks could be described within the framework of optimization and approximation techniques with dynamical systems.

Neural networks for preprocessing

A NN-based approach was introduced by Verikas and Bacauskiene [31] with salient features for classification. This is a feed-forward neural network. This approach involved neural network training with an augmented cross-entropy error function. A new feature selection algorithm [33] based on the wrapper approach using neural networks. The vital aspect of this algorithm is the automatic determination of neural network architectures during the feature selection process. According to this algorithm, it used a constructive approach involving correlation information in selecting features and determining neural network architectures. It would reduce the redundancy information resulting in compact neural network architecture.

Neural networks for classification

Jeich Mar et al. [34] proposed an IDS based on adaptive neuro-fuzzy inference system (ANFIS) rule to minimize the detection delay for the de-authentication attacks on the medium access control layer of a wireless local area network (WLAN). Both the average sequence number gap between the successive packets and the average statistical value of the de-authentication packets received by an access point are used to detect the de-authentication DoS attack. This ANFIS-IDS experimental platform was implemented and tested by the authors against real deauthentication DoS attack to empirically evaluate its average detection delay and average FAR. The performance of the IDS using the proposed ANFIS method was compared by them with non-parametric sequential change point detection algorithm in a practical WLAN environment.

Debar et al. [35] developed a NN model for IDS. The main advantage of this system is that the deviation to the normal behavior of the user could be easily diagnosed fairly and quickly by the NN. They used this capability as the goal of the IDS to detect potential intruders as quick as possible. Joo et al. [36] proposed a NN model to improve the performance of IDS using the asymmetric cost of false positive and false negative errors. Their approach differs from other approaches in measuring the system performance since it has considered asymmetric costs of errors rather than prediction accuracy in intrusion detection. Liu et al. [37] described a network IDS based on artificial neural networks (ANN). According to that system, the NNs are used to classify without consulting a domain expert; hence, this automation helped to detect both known and novel intrusions. The key part of the work was focused on the development of an adaptive resonance theory (ART) NN, and it is trained in real-time in an unsupervised way.

Moradi and Zulkernine [38] presented an ANN approach for intrusion detection. One of the limitations of that approach was the increase in training time, and also it does not provide a description on why certain network traffic was intrusive. Sarasamma et al. [39] proposed a novel multilevel hierarchical Kohonen net to detect intrusions in networks. In their work, randomly selected data points forming the KDD Cup 99 were used to train and test the classifier. The results obtained by them proved that the hierarchical Kohenen net in which each layer operates on a small subset of the feature space was superior to a Kohenen net operating on the entire feature space in detecting various kinds of attacks. Amini et al. [40] introduced an intelligent method for detecting known and unknown attacks. Unsupervised neural nets are used by them to detect intrusions in real time which can perform the analysis of new data over time without retraining. ART and self-organizing map NNs are evaluated using offline data in their work.

Koutsoutos et al. [41] presented a NN classifier ensemble system using a combination of NNs which is capable of detecting network attacks on web servers. Their system could identify unseen attacks and categorize them. The performance of the NN used by them for detecting attacks from audit dataset is fair with success rates of more than 78% in detecting novel attacks. However, it suffers from high false alarm rates; hence, it was necessary to propose suitable enhancements to the work. Shun and Malki [42] presented a NN-based IDS for detecting Internet-based attacks on a computer network. NNs are used by them to identify and predict current and future attacks in which the feed-forward NN with the back propagation training algorithm was employed to detect intrusions. They observed that the experimental results on KDD Cup 99 dataset show promising results for detection of intrusion when NNs are used for classification. Thomas and Balakrishnan [43] addressed the problem of optimizing the performance of IDS using fusion of multiple sensors. The trade-off between the detection rate and false alarm highlighted that the performance of the detector is better when the fusion threshold is less. In their work, NN-supervised learner has been designed and implemented to determine the weights of individual IDSs depending on their reliability in detecting a certain attack. The final stage of this data-dependent fusion architecture is a sensor fusion unit which computes the weighted aggregation in order to make an appropriate decision. The major limitation with this approach was that it required large computing power; hence, the training time was increased.

Linda et al. [44] presented an IDS using NN-based modeling for detection of anomalous activities. The major contributions of their approach are the use and analysis of real network data obtained from an existing critical infrastructure, the development of a specific window-based feature mining technique, construction of training dataset using randomly generated intrusion vectors, and the use of a combination of two NN learning algorithms, namely the error-back propagation and Levenberg-Marquardt algorithms, for normal behavior modeling. The major limitations of the approaches discussed in the literature using ANN for IDS is in training of NNs, i.e., computational load is very high. The time required for training is normally very high which is important for obtaining efficient NNs. Therefore, in the proposed work, a neuro-genetic algorithm has been designed and implemented by incorporating a genetic algorithm (GA) component into the ANN in the training phase. GA generates optimal weights by means of a special fitness function which has been designed specifically for weight adjustment in this research work; hence, it is used by ANN to learn the characteristics of normal pattern and attack types effectively.

Genetic algorithm-based IDSs

Genetic algorithm for simplified security audit trials analysis (GASSATA) proposed by Me [45], introduced a new genetic algorithm for the misuse intrusion detection. This GASSATA constructed a two-dimensional matrix. First, axis of the matrix specified different attacks which had been known already. Second, the axis represents different kinds of events derived from audit trails. Therefore, this matrix actually represented the patterns of intrusions. Given an audit record being monitored which had information about the number of occurrences of every event, this method applied genetic algorithms to find the potential attacks appearing in the audit record. However, the assumption that the attacks are dependent only on events in this method restricts its generality. There are two steps involved in genetic algorithm: one was coding a solution to the problem with a string of bits and the other was finding a fitness function to test each individual of the population against evaluation criteria.

Genetic-based feature selection

Most of the real life problems definitely need an optimal and acceptable solution rather than calculating them precisely at the cost of degraded performance, time and space complexities. Therefore, it was necessary to carry out the analysis using selected features. The problem of selecting significant features from KDD Cup 99 dataset for intrusion detection could not be represented in terms of formula since it was too complex. Moreover, when all the features are used without feature selection, it took a very longer time to calculate a solution precisely. Therefore, the feasible approach is to use a heuristic method which has performed the feature selection effectively. GA [46] was a heuristic, which stated that it estimated a solution and generated the optimized results. Among various heuristic methods, GA [47] was supposed to be more promising since it has differed in many ways from other heuristics. First, GA works on population of possible solutions, while other heuristic methods use a single solution in their iterations. Second, most heuristics are probabilistic or stochastic in nature and hence were not deterministic. On the other hand, each individual in the GA population contributes well to obtain a possible solution to the problem. In GA, the algorithm starts with a set of possible solutions represented by chromosomes called population. A potential solution to a specific problem was encoded in the form of a chromosome. By using the solution of one population, a new population is formed. Solutions are selected to form new solutions called offspring and are selected according to their fitness value. Finally, GA is more suitable in reducing the search space. Therefore, the convergence of the algorithm is faster when GA is employed. Subset generation [48] is a method of heuristic search, in which each instance in the search space specifies a candidate solution for subset evaluation. The decision process of this method is determined by some basic issues. Initially, the search starting point must be decided since it has controlled the direction of search. Feature selection search has started either with null set where features were added one by one or it was started with a full set of features and was eliminated one by one. But these methods have the drawback of being trapped into local optima [49].

Sindhu et al. [12] introduced a new intrusion detection model which was the combination of the following: (1) removing redundant instances in order to make the learning algorithm to be unbiased, (2) identifying suitable subset of features by employing a wrapper-based feature selection algorithm, (3) realizing proposed IDS with neuro tree to achieve better detection accuracy. The lightweight IDS has been developed using a wrapper-based feature selection algorithm that maximizes the specificity and sensitivity of the IDS as well as by employing a neural ensemble decision tree iterative procedure to evolve optimal features. An extensive experimental evaluation of the proposed approach with a family of six decision tree classifiers namely decision stump, C4.5, naive Baye's tree, random forest, random tree, and representative tree model to perform the detection of anomalous network pattern has been introduced by them.

Neuro-genetic classification

Genetic paradigm is employed to choose the predominant features, which has revealed the occurance of intrusions. The neuro-genetic IDS (NGIDS) involves calculation of weightage value for each of the categorical attributes so that data of uniform representation could be processed by the neuro-genetic algorithm. In this system, unauthorized invasion of a user were identified and newer types of attacks were sensed and classified respectively by the neuro-genetic algorithm. The experimental results obtained in this work shows that the system achieves improvement in terms of misclassification cost when compared with conventional IDS. The results of the experiments show that this system could be deployed based on a real network or database environment for effective prediction of both normal attacks and new attacks.

Fuzzy and rough sets

In this section, we discuss the topics namely fuzzy sets, neuro-fuzzy and rough-sets. Fuzzy logic would help to improve the detection accuracy when we are using different fuzzy logics. Rough sets could also be used to improve the detection accuracy.

Fuzzy sets

A Fuzzy multi-class support vector machine (SVM) was proposed in literature for network intrusion detection and is a collaborative intrusion detection model. Four kinds of SVM detection agents are discussed in their work and these agents have different attributes. They are used to detect transmission control protocol (TCP) attacks, UDP attacks, ICMP attacks, and content-based detection separately. A TCP detection agent was used as an example to illustrate the construction process of detection agent. This multi-agent collaborative detection method has increased the detection speed and accuracy. The intrusion detection based on fuzzy multi-class SVM has advantages in two aspects: (1) it selects the least attributes to build detection agents respectively and hence does not need all the attributes of the network packets; (2) it is a collaborative detection system which has improved not only the detection rate but also the detection accuracy. Du Hongle et al. [50] proposed an improved v-fuzzy support vector machine (FSVM) through introduction membership to each data point. They reformulate the improved v-FSVM so that different input points can make different contributions to decision hyperplane. In order to verify the performance of the improved v-FSVM, they applied it to intrusion detection.

Yu-Ping Zhou et al. [51] presented a hierarchical neuro-fuzzy inference intrusion detection system (HFIS). In their proposed system, principal component analysis neural network was used to reduce the input data space. An enhanced fuzzy c-means clustering algorithm was applied to create and extract fuzzy rules. The adaptive neural fuzzy inference system was utilized repeatedly in their model. At last, the system was optimized by genetic algorithm. The main advantages of the HFIS model are its capability to perform not only misuse detection but also anomaly detection. Moreover, their method has higher speed and better performance.

A hybrid intrusion detection method based on hidden Markov model (HMM) and fuzzy logic has been proposed by Li et al. [52]. The experimental results showed that their method is efficient to classify the anomaly profile from the normal profile. Comparing with other methods based on the HMM only, this HMM- and fuzzy logic-based method has the following advantages: first, it needs only less storage without the profile database. With the processes being used by more and more users, the profile database will be greatly enlarged. So, the profile database would occupy much storage with the larger and larger profile database; second, it could reduce training time effectively, which needed less testing data. When the profile databases are very large, the detection speed is slower as the sequence must be compared with all the records in the profile database. Their approach detects network-based attacks only at high false-positive rates as the processes in those attack scenarios behave similar to the normal behavior.

Neuro-fuzzy algorithms

Neuro-fuzzy algorithms are useful for classifying large volume of data with uncertainty. A novel neuro-fuzzy network for pattern classification problem has been proposed [53]. This flexible classification system is able to determine all of the parameters from the training set without any prior knowledge. The proposed classification model has been used in calculating the initial weights from the training data. This model contains two networks: one is the feature extraction unit and the other, the inference unit. The feature extraction unit effectively reduces the dimension of the original feature variables. The inference determines the classification results according to the distributions of the new feature variables.

An evolutionary soft computing approach for intrusion detection has been introduced by Toosi and Kahani [54] and has successfully demonstrated its usefulness on the training and testing subset of KDD cup 99 dataset. The ANFIS network was used as a neuro-fuzzy classifier for intrusion detection since ANFIS is capable of producing fuzzy rules without the aid of human experts. Also, subtractive clustering has been utilized to determine the number of rules and membership functions with their initial locations for better classification. Moreover, a fuzzy decision-making engine was developed to make the system more powerful for attack detection, using the fuzzy inference approach. At last, they proposed a method to use genetic algorithms to optimize the fuzzy decision-making engine. Yu-Ping Zhou et al. [55] presented a hierarchical neuro-fuzzy inference intrusion detection system. In the proposed system, princi pal-component-analysis neural network has been used to reduce the input data space. An enhanced fuzzy c-means clustering algorithm has been applied to create and extract fuzzy rules. The adaptive neural fuzzy inference system was utilized repeatedly in their model. At last the system has been optimized by genetic algorithm. The main advantage of their model is the capability to detect not only misuse but also anomaly. Moreover, their proposed method has higher speed and better performance.

Fuzzy-Genetic algorithms

A feature-extraction neuron-fuzzy classification model (FENFCM) has been proposed by Nai Ren Guo et al. [56] that enabled the extraction of feature variables and has provided the classification results. This classification model has been integrated with a standard fuzzy inference system and a neural network with supervised learning. The FENFCM automatically generated the fuzzy rules from the numerical data and triangular functions that were used as membership functions both in the feature extraction unit and in the inference unit. To adapt the proposed FENFCM, two modificatory algorithms are applied: first, they utilized evolutionary programming to determine the distribution of fuzzy sets for each feature variable of the feature extraction unit; second, the weight-revised algorithm is used to regulate the weight grade of the principal output node of the inference unit; finally, the FENFCM was validated using two benchmark data sets, the Wine database and the Iris database. Computer simulation results by them have demonstrated that the classification model provides a sufficiently high classification rate in comparison with that of other models proposed in the literature.

Rough sets

Zihui Che Xueyun Ji [56] presented a new anomaly detection model based on rough set reduction and HMM on the basis of the analysis of shortcomings of other detection methods these days. Specifically, that method has the following advantages:

1) The method of rough set reduction provided an efficient way to reduce the number of attributes, as well as the complexity of the information expression system. It has decreased the training time of HMM after the reduction of redundant information.
2) The rough set reduction process would also generate decision conditions, which could be applied to further detection after HMM evaluation. The strategy could revise the detection results to improve the accuracy of anomaly detection.
3) The HMM- and rough set-based approach could identify misuse and malicious intrusion by means of attribute reduction.

They could acquire a better HMM with a relatively small number of training data. Their method could promote the detection rate and decrease the false alarm rate stably.

A new feature selection algorithm combining a rough sets and genetic algorithms on the basis of clustering was proposed by Guo et al. [57]. Firstly, it uses the rough set theory to process selection and then uses the improved genetic algorithm based on clustering to find the optimal subset in the remaining subset. This algorithm has many advantages. In the end, it combines the results of the first two steps to get the final results. The results of the experiments show that this new method is better at detection accuracy and false rate than the other algorithms.

Particle swarm intelligence

Swarm intelligence approach

Honey bees exhibit many features that could be used as models for intelligent systems. These features include bee dance (communication), bee foraging, queen bee, task selection, collective decision making, nest site selection, mating, floral/pheromone laying and navigation systems.

Queen Bee Jung [58] proposed an evaluation method called queen-bee evolution simulating the queen bee role in the reproduction process. This method improved the optimization capability of genetic algorithms by enhancing exploitation and exploration processes. Xu et al. [59] developed a bee-swarm genetic algorithm for designing DNA sequences that satisfied some combinatorial and thermodynamic constraints, in which the optimum individual of population selected as a queen bee and a random population was introduced to reinforce the exploitation of genetic algorithm and increase the diversity of population.

Bee dance and communication Wedde et al. [60] presented a completely decentralized multi-agent approach on multiple layers where car or truck routing were handled through algorithms adapted from the BeeHive algorithms which in turn had been derived from honey bee behavior. They reported superior performance of over conventional approaches [61].

Task allocation Gupta and Koul [62] built an architecture named Swan based on the management of beehives by worker bees and the queen bee in the animal kingdom for network management of Internet protocol networks in order to overcome the shortcomings of traditional network management software. Similarity between honey bee and agents teamwork inspired Sadik et al. [63] proposed a system to develop a teamwork architecture to enhance the performance and task execution efficiency of software agents since a limited progress has been made towards efficient task execution mechanisms by group of agents in collaboration and coordination with each other. The authors named it Honey Bee teamwork architecture afterwards.

Collective decision and nest site selection Passino [64] established a mathematical model of the nest site selection process of honey bee swarms and highlighted the potential implications of the dynamics of swarm decision making. Gutierrez and Huhns [65] handled the quorum sensing during nest site selection in the area of design diversity of software fault tolerance.

Ant colony optimization

A novel approach for intrusion detection from the standpoint of feature selection was proposed by Gao et al. [66]. ACO was applied to select effective features for SVM classification. The simulation using KDD cup 99 dataset showed that SVM with obtained optimal feature subset could achieve better generalization performance than that without feature selection. This demonstrated the fact that dimension reduction could improve the generalization performance of intrusion detection and make the detection much more time efficient.

Rahul Karthik Sivagaminathan and Sreeram Ramakrishnan [67] presented a hybrid method based on ant colony optimization and ANNs to address feature selection. The proposed hybrid model was demonstrated by them using data sets from the domain of medical diagnosis, yielding promising results. An intrusion detection method based on ant colony fuzzy clustering has been proposed by Wei Song Li et al. [68]. The algorithm used the ant colony optimization algorithm which has a strong ability to deal with local minima since it is better than the random selected cluster centers that cause iterative process into a local optimal solution and dynamically determines the number and center of clusters. An efficient hybrid ant colony optimization-based feature selection algorithm has been presented by Md.Monirul Kabir et al. [69]. Since ants were the foremost strength of an ACO algorithm, guiding the ants in the correct directions was a critical requirement for high-quality solutions. Accordingly, this algorithm guided the ants during feature selection by determining the subset size. Furthermore, new sets of pheromone update and heuristic information measurement rules for individual features bring out the potential of the global search capability of this ACO-based feature selection algorithm.

Particle swarm optimization

A novel intrusion detection framework based on particle swarm optimization (PSO) was proposed by Jiang Tian and Gu [70] which had combined the idea of unsupervised learning method and the supervised strategy. Instead of calculating the accuracy, ROC analysis was utilized to evaluate the detection performance. This PSO algorithm has been executed for global optimal parameters of SVM. Best combination of TPR with FPR has been achieved after adjusting the offset of the detection function. The effectiveness of their method for anomaly detection was demonstrated on four benchmark datasets, and results have showed satisfactory performance.

Comparative analysis

Over the past decade, intrusion detection based upon computational intelligence approaches has been a widely studied topic, being able to satisfy the growing demand of reliable and intelligent intrusion detection systems.

In our view, these approaches contribute to intrusion detection in different ways. Fuzzy sets have represented and processed numeric information in a linguistic format, so they could make the system complexity manageable by mapping a large numerical input space into a smaller search space. In addition, the use of linguistic variables is able to present normal or abnormal behavior patterns in a readable and easy to comprehend format. The uncertainty and imprecision of fuzzy sets smooth the abrupt separation of normal and abnormal data, thus enhanced the robustness of an IDS.

Feature selection

Gradually feature removal method

The gradually feature removal (GFR) method [71] decides the importance of the 41 features of the KDD Cup dataset and gradually removes the less important features. This algorithm well and selects 19 features namely 2, 4, 8, 10, 14, 15, 19, 25, 27, 29, 31, 32, 33, 34, 35, 36, 37, 38, and 40. Using these 19 features, 98.6249% accuracy was achieved with SVM in tenfold cross validation. In order to evaluate the advantage of the GFR method, the other three feature reduction algorithms were also undertaken by the authors. In the feature removal method, 10 important features are chosen. Similarly, the sole feature method chooses other 10 critical features. Moreover, by choosing the common features selected by the two algorithms, 10 critical features are derived in the hybrid method.

Modified mutual information-based feature selection algorithm

The modified mutual information-based feature selection algorithm (MMIFS) was proposed by Fatemeh Amiri et al. [72]. Moreover, the authors have analyzed the features selected by their proposed MMIFS method and their relationship with different attack types. In the KDD Cup 99, dataset was used for carrying out the experiments detecting attacks. This algorithm selects features for identifying DoS, Probe, R2L, and U2R attacks effectively by computing the mutual information.

Mutual information based feature selection algorithm was initially proposed by Battiti [73] to maximize the relevance between the input features and the output and to minimize the redundancy of the selected features. The algorithm selects one feature at a time which maximizes the information with outputs. In this mutual information-based feature selection algorithm, the mutual information expression is adjusted by subtracting a quantity proportional to the average mutual information within the selected features. The main advantage of this algorithm is that it selects 13, 8, 15, and 10 features for Probe, DoS, R2L, and U2R which are optimal for classification.

CRF-based feature selection

Conditional random field (CRF)-based feature selection is a statistical approach proposed by Gupta et al. [74] for effective feature selection. They proposed a layered approach in which each layer considers one type of attack. Therefore, the probability value for each relevant feature is measured, and for each type of attack, different features are selected.

They used domain knowledge along with the practical significance, and they performed feasibility analysis for each feature before selecting it for a particular layer. Thus, from the total 41 features, they selected only 5 features for the Probe layer, 9 features for the DoS layer, 14 features for the R2L layer, and 8 features for the U2R layer. Since each layer is independent of every other layer, the feature sets for the layers are not disjoint.

Wrapper based genetic feature selection

In this model, genetic feature selection algorithm follows a wrapper-based approach. Moreover, each iteration of this algorithm results in a decision tree. After *n* iterations, a series of trees are obtained, and the best have been used to generate rules. The tree with the highest sensitivity and specificity are identified as the best trees. Thus, best set of features are extracted [75,76] based on sensitivity and specificity values. The main advantage of this genetic-based feature selection algorithm is that it selects only the important and contributing features for classification.

Comparison

The gradual feature removal method first removes the repeated data from the KDD cup dataset and uses k-means clustering to remove the next important set of features. However, the number of clusters is predetermined. In the modified mutual information-based feature selection algorithm, mutual information is used to perform feature selection and hence more flexible. In the CRF-based feature selection method, conditional probability values are used to select the relevant features and hence can handle uncertainty effectively. Finally, the wrapper-based method uses a decision tree to remove redundant subsets of features. However, from the analysis of all these methods, it is observed that the combination of mutual information and information gain ratio values provide a better method for feature selection since it can be used to perform both attribute selection and tuple reduction.

Classification

There are many works on classification that are available in the literature. Among them, the most relevant works for IDS are discussed in this section.

Linear programming system-based method for detecting U2R attacks

In this paper, a new approach for detecting U2R attacks has been investigated and evaluated. In their model, a behavior i belonging to an attack class j is represented by the variable x_{ij} and each class is represented by its feature vector as F_j. The distance between the behavior i and class j is represented as α_{ij}. The problem is represented for m attack classes and attack types contains β_j elements such that $\beta_j \geq 0$. Now, the problem is formulated using simplex model as

$$Z_{\min} = \sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}\, x_{ij}$$

$$x_{ij} \in \{0, 1\}$$

$$x_{ii} = 1$$

$$\sum_{j=1}^{m} x_{ij} = 1$$

$$\sum_{i=1}^{m} x_j \geq 0$$

They proposed an optimal algorithm to solve this problem, and based on that, they classified the attacks effectively.

Layered approach

We now describe the layer-based intrusion detection system (LIDS) proposed by Gupta et al. [74] in detail. According to them, the LIDS drew its motivation from the airport security model, where a number of security checks are performed one after the other in a sequence. Similar to this model, this LIDS represents a sequential layered approach was developed for ensuring availability, confidentiality, and integrity of data and (or) services over a network.

The major goal of using a layered model is to reduce computation time required to detect anomalous events and to improve the speed of operation of the system. In this approach, the algorithm uses the selected features and check whether there is a probe attack in the first layer called probe layer. Similarly, at each layer, it checks for the occurrence of the corresponding attacks. If there is an attack, it informs the prevention system. The main advantage of the layered approach is that it reduces the computation time by using separate feature selected by CRF-based feature selection algorithm.

Least squares support vector machine

SVM [77] is a supervised learning method used for solving classification and regression problems. An SVM can train with a large number of patterns. The least square support vector machine (LSSVM) is a modified algorithm [78] to the standard SVM. It solves a linear equation in the optimization stage and hence simplifies the process. Moreover, this LSSVM is effective since it avoids local minima in SVM problems used by LSSVM classifier is used by to detect normal and attacks data.

Neuro-tree classifier

In the neuro-tree classifier proposed by Sindhu et al. [12] for intrusion detection, the features selected by a genetic-based approach are used for classification. The major contributions of the neuro-tree classifier are the provisions of a new facility for the prevention of over fitting and the use of new fitness evaluation framework for maximizing the sensitivity and specificity. The main advantages of the neuro-tree classifier are that it reduces the false alarm rate and fast convergence.

Comparison

The linear programming system-based method used for classification is more efficient in detecting U2R attacks. The authors use the behavior distance between classes to find the similarity. However, it is necessary to focus on all types of attacks for providing effective security.

In the layered approach, each attack is analyzed in a separate layer and hence is effective in detecting all types of attacks. In the least square support vector machine-based classification uses an enhanced SVM to avoid local minima. This method detects all types of attacks with improved accuracy. The neuro-tree classifier provides effective classification when optimal features are provided. Hence, it reduces the false alarm rate effectively, and in addition, the algorithm converges fast.

Proposed intelligent IDS

In this paper, an intelligent IDS developed by proposing a new feature selection algorithm and a new classification algorithm is also discussed.

Feature selection

In this work, a new feature selection algorithm has been proposed by using an attribute selection and tuple selection. This algorithm has been proposed using rules and information gain ratio for attribute selection. In order to achieve this, the data set D is divided into n number of classes C_i. The attributes F_i having maximum number of nonzero values are chosen by the agent, and the information gain ratio is computed using Equations 1, 2, and 3, where F is the feature set.

$$\mathrm{Info}(D) = -\sum_{j=1}^{m}\left[\frac{\mathrm{freq}\left(C_j, D\right)}{\mid D\mid}\right]\log_2\left[\frac{\mathrm{freq}\left(C_j, D\right)}{\mid D\mid}\right] \tag{1}$$

$$\mathrm{Info}(F) = \sum_{i=1}^{n}\left[\frac{|F_i|}{\mid F|}\right] \times \mathrm{info}\ (F_i) \tag{2}$$

$$\mathrm{IGR}(A_i) = \left[\frac{\mathrm{Info}\ (D) - \mathrm{Info}\ (F)}{\mathrm{Info}\ (D) + \mathrm{Info}\ (F)}\right] \times 100 \tag{3}$$

In addition, tuple selection is also carried out using the rule-based approach.

Results and discussion

The agent-based attribute selection algorithm has been selected with 19 important features in Table 1. This selection was based on the information gain ratio values of various attributes.

Classification

In this paper, a new classification algorithm called IREMSVM algorithm has been proposed from the existing intelligent agent-based enhanced multiclass SVM (IAEMSVM) [79].

Enhanced multiclass support vector machine

In the IREMSVM algorithm, the data set is first divided into R classes. Then the distance between any two classes of patterns are computed from the R classes using the Minkowski distance. According this method, the distance between two points

$$P = (x_1, x_2, x_3, \dots, x_n) \text{ and}$$
$$Q = (y_1, y_2, y_3, \dots, y_n) \in R^n \tag{4}$$

is given by the formula given in Equation 5.

$$d_{ij} = \left(\sum_{i=1}^{n}\mid x_{ik} - x_{jk}\mid^{p}\right)^{\frac{1}{p}}, \tag{5}$$

where p is the order and it also finds the centroids of each class, where j and k are the neighbors of i.

The centroid is computed using the formula given in Equation 6:

$$C_i = \sum_{m=1}^{nt} X^{i}{}_{m}/n_i, \tag{6}$$

where C$_i$ = centroid value of i^{th} node, X = individual i^{th} lowest distance, and n = number of dimensions

The steps of this algorithm are as follows:

Algorithm 1 Intelligent rule-based attribute selection algorithm

```
Input: Set of 41 features from KDD'99 Cup data set
Output: Reduced set of features R
Steps of the algorithm:
Step 1:   Calculate the information gain ratio for each attribute $A_i$ ε $D$ using Equation 3.
Step 2:   Choose an attribute $A_i$ from $D$ with the maximum information gain ratio value.
Step 3:   Split the data set $D$ into subdatasets $\{D_1, D_2, \ldots, D_n\}$ depending on the attribute
    values of $A_i$ where $C_j$ stands for $j$th attribute of class $C$.
Step 4:   Find all the attributes whose information gain ratio > the threshold.
Step 5:   Store the selected attributes in the set R and output it.
Step 6:   Compute the mutual information value for each tuple.
Step 7:   Compare the key attribute values for each tuple with threshold.
Step 8:   If it is less than the threshold, then exit.
Step 9:   If key value is ≥ the threshold, then add the tuple into the ouput table.
```

Algorithm 2 Intelligent rule-based enhanced multiclass support vector machine

```
Step 1: Rule select two initial cluster centers by applying the intelligent rules.
Step 2: Import a new class C from the dataset.
Step 3: Compute the Minkowski distance between two classes.
Step 4: if (d (A, B) > d (A, C)) then
          B is assigned as Normal
          Else
          C is assigned as Attacker.
Step 5: Intelligent agent calculates the min and max of the distances.
Step 6: If (d (A, B) < threshold limit of the distance) then it creates a new cluster, and
this is the center of the new cluster.
          Else
          B is assigned as a suspect.
Step 7: Now compute mutual information value, and check it with a threshold.
Step 8: If it is the mutual information value ≥ threshold then
          Accept the record
          Else
          Reject the record
```

Experimental results

This work has been implemented using Java programs. Moreover, the experiments have been conducted to classify the KDD'99 Cup data set using both full features and selected features. So that comparative analysis can be performed.

Table 2 shows the comparison of SVM, IAEMSVM, and the proposed IREMSVM with respect to classification accuracy when the classification is proposed with the 19 selected features obtained from the proposed feature selection algorithms.

Table 1 List of 19 selected features

Selection number	Feature number	Feature name
1	2	protocol_type
2	4	src_byte
3	8	wrong_fragment
4	10	hot
5	14	root_shell
6	15	su_attempted
7	19	num_access_shells
8	25	rerror_rate
9	27	diff_srv_rate
10	29	srv_serror_rate
11	31	srv_diff_host_rate
12	32	dst_host_count
13	33	dst_host_srv_count
14	34	dst_host_same_srv_count
15	35	dst_host_diff_srv_count
16	36	dst_host_same_src_port_rate
17	37	dst_host_srv_diff_host_rate
18	38	dst_host_serror_rate
19	40	dst_host_rerror_rate

From this table, it is observed that the classification accuracy is increased in the proposed algorithm when it is compared with the existing algorithms for Probe, DoS, and others attacks. This is because the agents used in this proposed algorithm perform constraint checking for all types of experimental uses in the classification.

Conclusion

In this paper, a survey on intelligent techniques for feature selection and classification techniques used of Intrusion Detection has been presented and discussed. In addition, a new feature selection algorithm called Intelligent Rule based Attribute Selection algorithm and a novel classification algorithm named Intelligent Rule-based Enhanced Multiclass Support Vector Machine have been proposed. In this paper, intelligent algorithms for feature selection and classification have been proposed to design an effective intrusion detection system. The scope of this paper includes neural networks, fuzzy systems, genetic algorithm and particle swarm intelligence. The advantages and disadvantages of these intelligent techniques have been analyzed. The contributions of various research works in IDS are systematically summarized and compared, which allows us to clearly define existing research challenges and highlight promising new research directions. In addition the need for the new intelligent feature selection also called Intelligent Rule based Attribute Selection algorithm has been highlighted based on experimental results. In addition, the advantage of proposing the new classification also called Intelligent Rule-based Enhanced Multiclass Support Vector Machine has been discussed in detail so that the proposed system can be used to provide security to networks effectively.

Table 2 Detection accuracy comparisons with 19 features

Exp. No.	SVM			IAEMSVM			IREMSVM		
	Probe	DoS	Others	Probe	DoS	Others	Probe	DoS	Others
1	91.53	92.30	60.73	99.58	99.69	71.52	99.78	99.79	71.71
2	90.78	91.45	61.10	99.41	99.27	69.32	99.51	99.38	69.41
3	90.67	92.70	60.92	99.58	99.49	74.12	99.67	99.58	74.22
4	91.29	91.68	61.20	99.30	99.24	73.13	99.34	99.32	73.25
5	91.23	92.90	62.43	99.38	99.22	71.87	99.31	99.32	71.91

Abbreviations

ACO: Ant colony optimization; ANFIS: Adaptive neuro-fuzzy inference system; ART: Adaptive resonance theory; CRF: Conditional random field; DDoS: Distributed denial of service; DR: Detection rate; FAR: False alarm rate; FENFCM: Feature-extraction neuron-fuzzy classification model; FNR: False negative rate; FPR: False positive rate; FSVM: Fuzzy support vector machine; GA: Genetic algorithm; GASSATA: Genetic algorithm for simplified security audit trials analysis; GFR: Gradually feature removal method; HFIS: Hierarchical neuro-fuzzy inference system; HMM: Hidden Markov model; IAEMSVM: Intelligent agent-based multiclass support vector machine; IDS: Intrusion detection system; IREMSVM: Intelligent rule-based multiclass support vector machine; LIDS: Layer-based intrusion detection system; LSSVM: Least square support vector machine; MAN: Metropolitan area networks; MMIFS: Modified mutual information-based feature selection; NN: Neural network; PSO: Particle swarm intelligence; SVM: Support vector machine; TNR: True negative rate; TPR: True positive rate; WLAN: Wireless local area network.

Competing interests

The authors declare that they have no competing interests.

Received: 11 January 2013 Accepted: 30 October 2013
Published: 27 November 2013

References

1. W Stallings, *Cryptography and Network Security Principles and Practices* (Prentice Hall, Upper Saddle River, 2006)
2. J Anderson, *An Introduction to Neural Networks* (MIT, Cambridge, 1995)
3. B Rhodes, J Mahaffey, J Cannady, *Multiple self-organizing maps for intrusion detection, Paper presented at the Proceedings of the 23rd National Information Systems Security Conference, Baltimore, 16–19*, 2000
4. S Franklin, A Graser, Is it an agent or just a program? in *ECAI '96 Proceedings of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages* (Springer, London, 1996)
5. N Jaisankar, SGP Yogesh, A Kannan, K Anand, *Intelligent Agent Based Intrusion Detection System Using Fuzzy Rough Set Based Outlier Detection, Soft Computing Techniques in Vision Sci., SCI 395* (Springer, 2012), pp. 147–153
6. T Magedanz, K Rothermel, S Krause, *Intelligent agents: an emerging technology for next generation telecommunications? In INFOCOM'96 Proceedings of the Fifteenth Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, Mar 24–28*, 1996
7. I Guyon, S Gunn, M Nikravesh, L Zadeh, *Feature Extraction, Foundations and Applications* (Springer, Berlin, 2006)
8. R Kohavi, G John, Wrappers for feature subset selection. Artif Intell J Spec Issue Relevance **97**(1–2), 273–324 (1997)
9. P-N Tan, M Steinbach, V Kumar, *Introduction to Data Mining* (Addison-Wesley, Boston, 2005)
10. RO Duda, PE Hart, DG Stork, *Pattern Classification*, 2nd edn. (Wiley, Hoboken, 2000)
11. C Stefano, C Sansone, M Vento, To reject or not to reject: that is the question: An answer in the case of neural classifiers. IEEE Trans. Syst. Manag. Cyber **30**(1), 84–94 (2000)
12. SS Sivatha Sindhu, S Geetha, A Kannan, Decision tree based light weight intrusion detection using a wrapper approach. Expert Syst. Appl **39**, 129–141 (2012)
13. A Ghadiri, N Ghadiri, An adaptive hybrid architecture for intrusion detection based on fuzzy clustering and RBF neural networks, in *Proceedings of the 2011 Ninth IEEE Conference on Annual Communication Networks and Services Research Conference, Otawa* (IEEE Computer Society, Washington, 2011), pp. 123–129
14. S Chebrolu, A Abraham, P Johnson, Thomas Feature deduction and ensemble design of intrusion detection systems. Computers & Security **24**(4), 295–307 (2005)
15. W Zhang, S Teng, H Zhu, H Du, X Li, *Fuzzy Multi-Class Support Vector Machines for Cooperative Network Intrusion detection. Proc. 9th IEEE Int. Conference on Cognitive Informatics (ICCI'10)* (IEEE, Piscataway, 2010), pp. 811–818
16. L Zadeh, Role of soft computing and fuzzy logic in the conception, design and development of information/intelligent systems, in *Computational Intelligence: Soft Computing and Fuzzy-neuro Integration with Applications*, ed. by O Kaynak, L Zadeh, B Turksen, I Rudas. Proceedings of the NATO Advanced Study Institute on Soft Computing and its Applications held at Manavgat, Antalya, Turkey, 21–31 August 1996, volume 162 of NATO ASI Series (Springer, Berlin, 1998), pp. 1–9
17. M Tavallaee, E Bagheri, W Lu, AA Ghorbani, A Detailed Analysis of the KDD CUP 99 Data Set, in *Proceedings of the IEEE Symposium on Computational Intelligence in Security and Defense Applications* (Ottawa, 2009)
18. KDD, KDD Cup, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, Accessed October 2007
19. SJ Stolfo, W Fan, W Lee, A Prodromidis, PK Chan, Cost-Based Modeling for Fraud and Intrusion Detection: Results From the JAM Project, in *Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX '00), Hilton Head, SC*, 2000
20. RP Lippmann, DJ Fried, I Graf, JW Haines, KR Kendall, D McClung, D Weber, SE Webster, D Wyschogrod, RK Cunningham, MA Zissman, *Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation, Hilton Head, 25–27 January 2000, vol 2* (IEEE, Amsterdam, 2000), pp. 10–12
21. MIT Lincoln Labs, *DARPA Intrusion Detection Evaluation*, 1998. http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/index.html, Accessed February 2008
22. AA Bakar, ZA Othman, AR Hamdan, R Yusof, R Ismail, *An Agent Based Rough Classifier for Data Mining. Eighth International Conference on Intelligent Systems Design and Applications , vol 1* (IEEE Computer Society, Washington, 2008), pp. 145–151
23. A Fagiolini, G Valenti, L Pallottino, G Dini, A Bic, Decentralized Intrusion Detection for Secure Cooperative Multi–Agent Systems, in *Proceedings of the 46th IEEE Conference on Decision and Control* (IEEE, Amsterdam, 2007), pp. 1553–1558
24. X Zhu, Z Huang, H Zhoul, *Design of a Multi-agent Based Intelligent Intrusion Detection System. IEEE International Symposium on Pervasive Computing and Applications* (IEEE, Amsterdam, 2006), pp. 290–295
25. G Xiantai, J Weidong, Z Dao, Multi Agent System for Detection and Containment in Metropolitan Area Networks. J. Electron. (China) **23**(2), 259–265 (2006)
26. N Joukov, *C T-c, Internet worms as internet-wide threat*. http://www.ecsl.cs.sunysb.edu/tr/TR143nikolaiRPE.pdf, Accessed Sept 2003
27. G Bourkache, M Mezghiche, K Tamine, A Distributed Intrusion Detection Model Based on a Society of Intelligent Mobile Agents for Ad Hoc Network, in *the 2011 Sixth IEEE International Conference on Availability, Reliability and Security, Vienna, August 2011* (IEEE, Amsterdam, 2011), pp. 569–572
28. Y Wang, S Behera, J Wang, G Helmer, V Honavar, L Miller, R Lutz, M Slagell, Towards the automatic generation of mobile agents for distributed intrusion detection system. J. Syst. Softw. **1**(34), 1–14 (2006). Elsevier
29. C-h Fonk, GP Parr, PJ Morrow, Security schemes for Mobile Agent based Network and System Management Framework. J. Networks Syst. Manag. Springer **19**, 232–256 (2011)

30. P Mell, D Marks, M McLarnon, A Denial of service resistant intrusion detection architecture. Comput Networks J Elsevier, Amsterdam, 2000
31. A Verikas, M Bacauskiene, Feature selection with neural networks. Pattern Recognition Letters, Elsevier **23**, 1323–1335 (2002)
32. C-H Tsang, S Kwong, Multi-Agent Intrusion Detection System in Industrial Network using Ant Colony Clustering Approach and Unsupervised Feature Extraction, in *the IEEE Conf. Proc. on Industrial Technology* (IEEE, Amsterdam, 2005), pp. 51–56
33. MM Kabir, MM Islam, K Murase, A New Wrapper Feature selection approach using Neural Network. Neuro Computing **73**, 3273–3283 (2010). Elsevier
34. J Mar, Y-C Yeh, I-F Hsiao, *An ANFIS-IDS against Deauthentication DOS Attacks for a WLAN Taichung, 17–20 October 2010* (IEEE, Amsterdam, 2010), pp. 548–553
35. H Debar, M Becker, D Siboni, A neural network component for an intrusion detection system, in *IEEE Symposium on Research in Computer Security and Privacy, Oakland, 4–6 May 1992* (IEEE, Amsterdam, 1992), pp. 240–250
36. D Joo, T Hong, I Han, The neural network models for IDS based on the asymmetric costs of false negative errors and false positive errors. Expert Syst. Appl **25**, 69–75 (2003)
37. Y Liu, D Tian, A Wang, *ANNIDS: intrusion detection system based on artificial neural network, In Proceedings of the Second International Conference on Machine Learning and Cybernetics*, vol. 3 (IEEE, Amsterdam, 2003), pp. 2–5
38. M Moradi, M Zulkernine, *A neural network based system for intrusion detection and classification of attacks, in Proceedings of IEEE International Conference on Advances in Intelligent Systems – Theory and Applications, Luxembourg*, vol. 148 (IEEE, Amsterdam, 2004), pp. 1–6
39. S Sarasamma, Q Zhu, J Huff, Hierarchical Kohonen net for anomaly detection in network security. IEEE Transactions on System, Man, Cybernetics, Part B, Cybernetics **35**(2), 302–312 (2005)
40. M Amini, R Jalili, HR Shahriari, RT-UNNID: A practical solution to real-time network-based intrusion detection using unsupervised neural networks. Computers and Security Science Direct **25**(6), 459–468 (2006)
41. S Koutsoutos, IT Christou, S Efremidis, A classifier ensemble approach to intrusion detection for network-initiated attacks, in *Proceedings of the International Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, vol. 160 (IOS, Amsterdam, 2007), pp. 307–319
42. J Shun, HA Malki, Network intrusion detection system using neural networks. Proc. Fourth IEEE Int Conf Nat Comput **5**, 242–246 (2008). ICNC'08
43. C Thomas, N Balakrishnan, Improvement in intrusion detection with advances in sensor fusion. IEEE Trans. Inf Forensics Secur **4**(3), 542–551 (2009)
44. O Linda, T Vollmer, M Manic, Neural network based intrusion detection system for critical infrastructures, in *Proceedings of IEEE International Joint Conference on Neural Networks, Georgia* (IEEE, Amsterdam, 2009), pp. 102–109
45. L Me, GASSATA, a genetic algorithm as an alternative tool for security audit trials analysis, in *Proceedings of 1st International workshop on Recent Advances in Intrusion Detection* (Belgium, 1998)
46. DE Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning* (Addison-Wesley, Boston, 1989)
47. G Stein, B Chen, AS Wu, KA Hua, *Decision tree classifier for network intrusion detection with GA-based feature selection, Proceedings of the 43rd Annual Southeast Regional Conference*, vol. 2 (ACM, Georgia, 2005), pp. 136–141
48. R Curry, P Lichodzijewski, MI Heywood, Scaling genetic programming to large datasets using hierarchical dynamic subset selection. IEEE Trans. Syst. Man Cybern. B Cybern. **37**(4), 1065–1073 (2007)
49. J Doak, *An evaluation of feature selection methods and their application to computer security, Technical Report* (University of California at Davis, Department of Computer Science, 1992)
50. D Hongle, T Shaohua, Z Qingfang, *Intrusion detection based on fuzzy support vector machines. International Conference on Networks Security, Wireless Communications and Trusted Computing, vol 2* (IEEE Computer Society, Washington, 2009), pp. 639–642
51. Y-P Zhou, J-A Fang, Y-P Zhou, *Intrusion Detection Model Based on Hierarchical Fuzzy Inference System. Second IEEE International Conference on Information and Computing Science , vol 2* (IEEE Computer Society, Washington, 2009), pp. 144–147
52. Y Li, R Wang, J Xu, G Yang, B Zhao, *Intrusion detection method based on fuzzy hidden Markov model. Sixth IEEE International Conference on Fuzzy Systems and Knowledge Discovery, vol 3* (IEEE, Piscataway, 2009), pp. 470–474
53. NR Guo, T-HS Li, Construction of a neuron-fuzzy classification model based on feature-extraction approach. Expert Syst. Appl **38**, 682–691 (2011)
54. AN Toosi, M Kahani, A new approach to intrusion detection based on an evolutionary soft computing model using Neuro-fuzzy classifiers. Comput. Commun. **30**, 2201–2212 (2007)
55. Y-P Zhou, J-A Fang, Y-P Zhou, *Research on Neuro-Fuzzy Inference System in Hierarchical Intrusion Detection. IEEE International Conference on Information Technology and Computer Science* (IEEE Computer Society, Washington, 2009), pp. 253–256
56. ZCX Ji, *An efficient intrusion detection approach based on hidden Markov model and rough set. IEEE International Conference on Machine Vision and Human-machine Interface* (IEEE Computer Society, Washington, 2010), pp. 476–479
57. Y Guo, B Wang, X Zhao, X Xie, L Lin, Q Zhou, *Feature Selection Based on Rough Set and Modified Genetic Algorithm for Intrusion Detection. IEEE International Conference on Computer Science & Education* (IEEE, Piscataway, 2010), pp. 1441–1446
58. SH Jung, Queen-been evaluation for genetic algorithms. Electron. Lett. **36**(6), 575–576 (2003)
59. C Xu, Q Zhang, J Li, X Zhao, *A bee swarm genetic algorithm for the optimization of dna encoding. 3rd International Conference on Innovative Computing Information and Control. 35* (IEEE, Piscataway, 2008)
60. H Wedde, S Lehnohoff, B van Bonn, Z Bay, S Becker, S Bottcher, C Brunner, A Buscher, T Furst, A Lazagrescu, E Rotaru, S Senge, B Steinbach, F Yilmaz, T Zimmermann, *A novel class of multi-agent algorithms for highly dynamic transport planning inspire by honey bee behavior. IEEE conference on emerging technologies and factory automation* (IEEE, Piscataway, 2007), pp. 1157–1164
61. H Wedde, S Lehnohoff, B van Bonn, Z Bay, S Becker, S Bottcher, C Brunner, A Buscher, T Furst, A Lazagrescu, E Rotaru, S Senge, B Steinbach, F Yilmaz, T Zimmermann, *Highly dynamic and adaptive traffic congestion avoidance in real time inspired by honey bee behavior. Mobilitat and Echtzeit, Informatik aktuell, 21–31* (Springer, Berlin, 2008)
62. A Gupta, N Koul, SWAN: a swarm intelligence based framework for network management of ip networks. Int Conf Comput Intell Multimedia Appl **1**, 114–118. IEEE Computer Society, Washington, 2007
63. S Sadik, A Ali, HF Ahmed, H Suguri, *Honey bee teamwork architecture in multi-agent systems", Computer supported cooperative work in design III, Lecture notes in computer science, 4402/2007* (Springer, Berlin, 2007), pp. 428–437
64. K Passino, *Systems biology of group decision making. 14th Mediterranean conference on control and automation* (IEEE Computer Society, Washington, 2006)
65. RLZ Gutierrez, M Huhns, *Multi agent based fault tolerance management for robustness In Robust Intelligent Systems* (Springer, Berlin, 2008), pp. 23–41
66. H-H Gao, H-H Yang, X-Y Wang, *Ant Colony Optimization Based Network Intrusion Feature Selection And Detection. Proc. Fourth International Conference on Machine Learning and Cybernetics* (Springer, Berlin, 2005), pp. 3871–3875
67. RK Sivagaminathan, S Ramakrishnan, A hybrid approach for feature subset selection using neural networks and ant colony optimization. Expert Syst. Appl **33**, 49–60 (2007)
68. WS Li, XM Bai, LZ Duan, X Zhang, *Intrusion Detection based on ant colony algorithm of Fuzzy clustering. International Conference on Computer Science and Network Technology* (IEEE, Piscataway, 2011), pp. 1642–1645
69. MM Kabir, M Shahjahan, K Murase, A new hybrid ant colony optimization algorithm for feature selection. Elsevier-Expert Syst. Appl **39**, 3747–3763 (2012)
70. J Tian, H Gu, *Anomaly detection combining one-class SVMs and Particle swarm optimization algorithms, Nonlinear Dynamics*, vol. 61 (Springer, Berlin, 2010), pp. 303–310
71. Y Li, J Xia, S Zhang, J Yan, X Ai, K Dai, An efficient intrusion detection system based on support vector machines and gradually feature removal method. Expert Syst. Appl **39**, 424–430 (2012)
72. F Amiri, MMR Yousefi, C Lucas, A Shakery, N Yazdani, Mutual information-based feature selection for intrusion detection systems. J. Network Comput. Appl **34**, 1184–1199 (2011)
73. R Battiti, Using mutual information for selecting features in supervised neural net learning. IEEE Trans. Neural Netw. **5**, 537–550 (1994)
74. KK Gupta, B Nath, R Kotagiri, Layered Approach using Conditional Random Fields for Intrusion Detection. IEEE Trans. Dependable Secure Comput **7**, 1 (2010)
75. S Benferhat, K Tabia, *On the combination of Naive Bayes and decision trees for intrusion detection, IEEE International Conference on Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, vol. 1* (Piscataway, IEEE, 2006), pp. 211–216
76. H Liu, L Yu, Toward integrating feature selection algorithms for classification and clustering. IEEE Trans. Knowl. Data Eng. **17**, 491–502 (2005)

77. C Cortes, V Vapnik, Support vector networks. Mach. Learn. **20**, 1–25 (1995)
78. JAK Suykens, L Lukas, DP Van, MB De, J Vandewalle, Least squares support vector machine classifiers: a large scale algorithm, in *Proceedings of the European Conference on Circuit Theory and Design*, 1999, pp. 839–842
79. S Ganapathy, P Yogesh, A Kannan, Intelligent Agent based Intrusion Detection using Enhanced Multiclass SVM. Comput. Intell. Neurosci. **2012**, 10 (2012)

doi:10.1186/1687-1499-2013-271
Cite this article as: Ganapathy *et al.*: **Intelligent feature selection and classification techniques for intrusion detection in networks: a survey.** *EURASIP Journal on Wireless Communications and Networking* 2013 **2013**:271.