A Temporal Oriented Intelligent Genetic Neural Network Model for Effective Intrusion Detection

Rm.Somasundaram[1], K.Lakshmanan[2], V.K.Shunmuganaathan[3]
[1]Computer Applications, [2]Computer Engineering, [3]Mechanical Engineering
[1]SNS College of Engineering, Coimbatore, India,
[2]Sri Durgadevi Polytechnic College, Chennai, India,
[3]SNS College of Engineering, Coimbatore, India
[2]lakshkl@yahoo.com

***Abstract* - Security is an important challenge in internet based communication. In such a scenario, intrusion detection systems help to secure the data through the identification of normal and abnormal behaviors. In order to model these behaviors accurately and to improve the performance of the intrusion detection system, a temporal oriented heuristic genetic neural network (THGNN) is proposed in this paper. In this model, feature selection, structure design and weight adaptation are jointly in considered to analyze the interdependence of input features which helps to modify the network structure and connection weights. Moreover, the genetic algorithms are proposed to work with input nodes and hidden nodes. The crossover operator based on temporal constraints are introduced and considering the relationship between genotype and phenotype. Moreover, a temporal logic based adaptive mutation rate is applied, and the mutation operation is performed heuristically from time based weight adaptation, node manipulation. When the population is not evolved continuously for a time interval, the mutation rate is increased and the mutation type is changed. This temporal heuristic approach helps to perform weight adjustment effectively. Experimental results obtained using the KDD-99 dataset show that the proposed THGNN achieves better detection accuracy in terms of increased detection rate and decreased false positive rate.**

***Keywords* - intrusion detection, neural network, genetic algorithm, mutation operator, penalty factor, temporal constraints**

I. INTRODUCTION

With the growth of Internet applications and frequent network transmissions, traditional intrusion detection is not able to adopt high-speed and large-scale network communication environment. Therefore it is necessary to detect intrusions efficiently and at the same time to secure the normal network applications that use internet for data communication.

Intrusion Detection systems [1] identify those users (such as hackers) who are not authorized to use the computer system, and the individual’s software’s and data (e.g. internal attack) where though they are legally authorized may try to abuse their authority. An intrusion detection system can be viewed as a classification problem which divides the user behaviors into normal and abnormal behaviors based on their network usage. Therefore, the intrusion detection problem can be considered as a data mining application. Recently, a number of classification technologies are used in intrusion detection [1]. It includes decision trees, neural networks, genetic algorithms, Bayesian classification, hidden Markov models, rough sets, fuzzy sets, support vector machines and so on. Based on the application of these classification techniques a number of new intrusion detection systems have been proposed in the past.

Recently, machine learning algorithms namely rule learning, hidden Markov model, Support Vector Machine and neural network [1],[3]-[5] have been used in the field of intrusion detection. Moreover, neural networks have the capability of generalized from limited, noisy and incomplete information, an intrusion detection system developed using neural network can recognize not only previously known attacks but also novel attacks [1]. Therefore, the neural network is a promising technique for intrusion detection, when it is combined with a suitable weight adjustment method.

Feature selection [10]-[12] is one of the common tasks used in classification. It is used to reduce the number of inputs to a classification algorithm. Many techniques are already available for classification. Since feature selection helps to reduce the number of attributes that are considered when building a classification model, it is necessary to focus on both feature selection and classification.

In this paper, we propose a new Temporal oriented Heuristic Genetic Neural Network (THGNN) for effective intrusion detection. The main advantage of this temporal oriented intrusion detection model is ability to detect both known and novel attacks. The rest of this paper is organized as follows: Section 2 provides the literature survey. Section 3 describes the overall system architecture. Section 4 explains the proposed work. Section 5 depicts the result and discussions. Section 6 gives the conclusions and future enhancement.

II. LITERATURE SURVEY

There are many works on neural network based IDS. Among them, a heuristic genetic neural network was proposed by Biying Zhang [1] to improve the performance of intrusion detection, in which input features, network structure and connection weights are evolved jointly. Their experimental results show that the HGNN accomplishes feature selection, structure optimization and weight adaptation effectively. Through the comparative analysis, the authors have proved HGNN achieved better detection performance when it is compared with structure of neural network.

A collaborative intrusion detection model based on multi-class SVM is proposed by Wei Zhang et al [2]. In their models, fuzzy multi-class SVM is used for network intrusion detection. The architecture of their detection model and functions of every component are described in their paper. Four kinds of SVM detection agents are constructed by them in which the agents have different attributes. Separate agents are used to detect TCP, UDP and ICMP attacks and also for content-based detection. The TCP detection agent is used as an example to illustrate the construction process of detection agent. This approach based on a multi-agent collaborative detection increases detection speed and accuracy for handling network attacks. Therefore, their model is not only improves the detection rate, but also detection accuracy is markedly improved.

Koutsoutos et al [3] 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 also by categorizing 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 is necessary to propose suitable enhancements to their work.

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 in [4]. This flexible classification system is able to determine all of the parameters from the training set without any prior knowledge. This classification model has been used for calculating the initial weights from the training data. Moreover, this model contains two networks in which one is to feature extraction and the other for the inference. The feature extraction unit effectively reduces the dimension of the original feature variables. The inference unit determines the classification results according to the distributions of the new feature variables.

Ganapathy et al [5] proposed a new intrusion detection system called IAFSHC for securing MANETs effectively. This system has been developed by combining intelligent agent-based weighted distance outlier detection (IAWDBOD) algorithm with another algorithm called intelligent agent-based enhanced multiclass support vector Machine (IAEMSVM) algorithm. Moreover, an effective preprocessing technique called intelligent agent-based attribute selection algorithm (IAASA) was also proposed in their model which is included in IAFSHC to improve the detection accuracy and also to reduce the processing time.

Huaguang Zhang et al [6] proposed a new algorithm for pattern classification using a new Data Core based Fuzzy Min-Max Neural Network. However, it is necessary for the hyperbox to be expanded so that it can overlap with the previous hyperboxes. Moreover, it generates minimum number of hyperboxes for rule extraction. Oong et al [7] presented a novel replacement evolutionary approach known as the Hybrid Evolutionary Artificial Neural Network (HEANN) for simultaneously evolving a man-made neural networks (ANNs) topology and weights. Gui and Qiao [8] outlined a learning algorithm for nonlinear modeling and classification with that of radial basis operate neural networks to quicken the learning speed and optimization method for the RBF neural networks. Though all these works considered the classification of user data, real time intrusion detection can be effective, if it considers temporal phenomena. Hence, a temporal oriented classification system based on intelligent heuristic neural network is proposed in this paper for effective intrusion detection.

III. SYSTEM ARCHITECTURE

This intrusion detection system consists of eight components namely, KDD Cup Dataset, User Interface Module, Temporal Fitness Evaluation, Subnet Crossover Operator, Heuristic Mutation Operator, Temporal decision agent, rule base and Decision Intrusion module. The System Architecture is shown in figure 1.

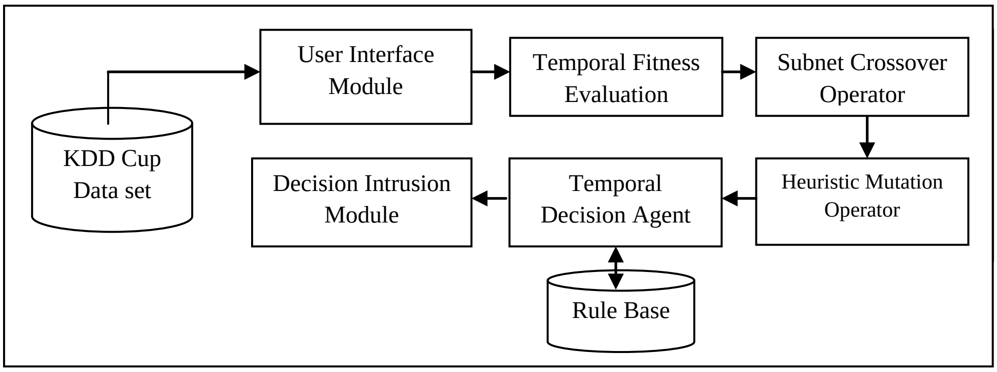
KDD Cup Data set
User Interface Module
Temporal Fitness Evaluation
Subnet Crossover Operator
Heuristic Mutation Operator
Temporal Decision Agent
Decision Intrusion Module
Rule Base

Fig. 1. System Architecture

The user interface module collects the important data from KDD'99 Cup dataset. Fitness evaluation component evaluates the fitness values. Subnet Crossover operator component applies the crossover operation. Heuristic Mutation Operator component applies the mutation operation for the features. Temporal decision agent component applies the temporal constraints for the features. The decision intrusion module decides whether the particular data is normal or abnormal. Rule base contains the rules.

IV. PROPOSED WORK

This section explains the proposed model is called Temporal oriented Intelligent Heuristic Genetic Neural Network (TIHGNN). The five important components of the proposed model are namely the temporal fitness evaluation, Subnet Crossover Operator, Heuristic Mutation Operator temporal constraint analysis and the overall evolutionary frame work in the following four subsections.

A. Temporal Fitness Evaluation

To evolve input features and network structure effectively, this work introduces a new temporal fitness function. Moreover, this function increases the detection accuracy rate and includes penalty factors for the number of input and hidden nodes. The temporal fitness function of the individual *a* at an interval is defined by

$$fit(a, t1, t2) = drate(a, t1, t2) \times \psi(\mathrm{a}, \mathrm{t1}, \mathrm{t2}) \times \varphi(a, t1, t2) \qquad (1)$$

where *drate* is the detection accuracy rate, ψ is the penalty factor for the number of input nodes, φ is the penalty factor for the number of hidden nodes and $[t_1,t_2]$ is the time interval.

The detection accuracy rate of the individual *a* at the time interval $[t_1,t_2]$ is defined by

$$drate(a, t1, t2) = correct(a, t1, t2)/sum(a, t1, t2) \qquad (2)$$

where *correct* is the number of accurate detections, *sum* is the total number of detections which include both normal instances and abnormal instances.

The penalty factor ψ is defined by

$$\psi(a, t1, t2) = 1 - (c_i n(a, t1, t2) - m_i, t1, t2) \times p_i n \qquad (3)$$

where $c_i n$ is the number of input nodes, m_i is the minimum number of input nodes, $p_i n$ is a very small user-defined parameter which is used to control the influence of the number of input nodes on fitness evaluation.

The penalty factor φ is defined by

$$\varphi(a, t1, t2) = 1 - (c_h(a, t1, t2) - m_h) \times p_h \qquad (4)$$

where c_h is the number of hidden nodes, m_h is the minimum number of hidden nodes, p_h is a very small user-defined parameter which is used to control the influence of the number of hidden nodes on fitness evaluation.

B. Subnet Crossover Operator

The generated subnet *Gen*(*i*) of the node *i* is defined by

$$Gen(i) = \begin{cases} CO(i) & i \in M \\ CI(i) \cup CO(i)Ui & i \in H \\ CI(i) \cup i & i \in P \end{cases} \qquad (5)$$

where *M* is the set of input nodes, *H* is the set of hidden nodes, *P* is the set of output nodes, *CI*(*i*) is the set of all input connections of the node *i*, *CO*(*i*) is the set of all output connections of the node *i*. The generated subnet of an input node is the set of its all output connections. The generated subnet of a hidden node is the set of itself, it's all input and output connections.

The generated subnet of an output node is the set of itself and its all input connections. The connection information includes whether or not the connection exists and the connection weight value. The node information includes bias and activation function.

In this paper, the subnet crossover operator is employed in consideration of the relationship between genotype and phenotype. The overall procedure of crossover operator is described as follows:

1. Select randomly a node $i \in M \cup H \cup P$.

2. Find the cross over for the selected node i.

C. Heuristic Mutation Operator

1) Adaptive mutation rate

Adaptive mutation rate is defined by

$$P(g) = (Gen + 1) * u \qquad (6)$$

where g denotes the current generation, *Gen* denotes the maximum number of generations for which the population is not optimized continuously, u denotes a user-defined parameter which is used to control the increase of the mutation rate.

If there are better solutions at each generation than the previous generation, *Gen* is equal to 0, and $P(g)$ is equal to 0.05. It can be considered that the crossover operator is working efficiently, and the mutation operator is not necessary for evolution. The more generations for which the population is not be enhanced, the larger *Gen* is. This results in the increase of mutation rate $P(g)$. Here, as the mutation rate increases, it can be considered that the evolutionary procedure by the crossover operator may be trapped into local minima and it becomes more difficult to search the optimal solution only by the crossover operator, and the mutation operator should be used to extend the space of solutions.

2) Heuristic selection of mutation operations

The mutation operator is composed of three operations: weight adaptation, node deletion and node addition. Weight adaptation is implemented by perturbing the weights of neural network using random noise, which is described by

$$w = w + N(0, a.P(g)) \qquad (7)$$

where $N(0,a \;.\; P(g))$ is a random variable with mean 0 and standard deviation $P(g)$, w is a weight, $P(g)$ is the adaptive mutation rate defined in (6), a is a user defined constant used to scale $P(g)$.

Node deletion means setting all connections on the deleted node to 0, and node addition means assigning connection weights between the added node and the other relative nodes. Under the encoding scheme in this paper, deleting a node is to set the corresponding columns and rows to 0, and adding a node is to assign random values to the corresponding columns and rows. The mutation operator is defined in equation (8).

$$Mutation\ Operator = \begin{cases} WA & P(g) < M \\ ND & M \le P(g) < N \\ NA & N \le P(g) \end{cases} \qquad (8)$$

where *WA* denotes the adaptation of connection weights, *ND* denotes the node deletion, *NA* denotes the node addition, $P(g)$ is the adaptive mutation rate defined in (6), M and N are the user-defined parameter which satisfy $0<M<N<1$. Obviously, which mutation operation is performed depends highly the mutation rate, and mutation rate is determined by the maximum number of generations the populations do not get better. Hence, the adaptive mutation rate given in (6) is the foundation of the total mutation operator. When the subnet crossover operator is working efficiently and improving the performance of the neural network continuously, *Gen* and $P(g)$ are very small, the weight adaptation will be performed. As the efficiency of the crossover operator becomes worse, *Gen* and $P(g)$ becomes larger, the node deletion and addition will be carried out.

The node deletion has more priority than the node addition to be apt to smaller network structure.

3) The heuristic mutation procedure

The heuristic mutation procedure is described as follows:

```
Calculate the adaptive mutation rate P(g)
While ( each individual in the population)
  If (U(0,1)< P(g))
        Select a mutation operation from weight adaptation, node deletion and node addition according to
        the equation (8)
  If (weight adaptation is selected)
        Uniformly select a connection weight and adapt the selected weight with equation (7)
```

```
    Else If (node deletion is selected)
          Uniformly select a node, not all connection weights on which is 0
          Set all connection weights to 0 between the node and the other nodes
    Else If (node addition is selected)
          Uniformly select a node, all connection weights on which is 0
         Assign the random values between -1 and 1 to the connections between the node
          and the other nodes
    End If
  End If
End while
```

D. Overall evolutionary framework

The initial population of the neural networks is generated with random weights and full connection. First, all initial or selected individuals are evaluated with the fitness function, and then, the best ones are selected from the parent and child individuals. Subsequently, the subnet crossover operator is performed on the selected individuals. Finally, the proposed heuristic mutation operator is carried out.

The overall evolutionary procedure is described as follows:

1. **Initialize:** Randomly generate an initial population of neural networks.
2. **Evaluation:** Evaluate all individuals of the initial population according to the fitness function.
3. **While** (Stopping conditions are not satisfied)
4. **Selection:** Select best *n* individuals from all parent and child individuals according to the fitness function.
5. **Crossover:** Perform the subnet crossover operator on the selected individuals.
6. **Mutation:** Perform the heuristic mutation operator on the selected individual.
7. **Evaluation:** Evaluate all evolved individuals.
8. Apply **temporal** constraints
9. **Select** the best population for the time interval [t_1, t_2]
10. **End While**

V. RESULTS AND DISCUSSION

A. Datasets

The experiments were performed on the KDD Cup 1999 dataset to assess the effectiveness of the proposed Temporal oriented Intelligent Heuristic Genetic Neural Network (TIHGNN). The dataset is classified into five major categories: Normal, denial of service (DOS), remote to local (R2L), user to root (U2R) and Probe. A 10% of data selection from training dataset is used to train neural network.

B. Experimental Results

Table I shows the comparison of Constrained JENN, JENN, HGNN and proposed TIHGNN with respect to detection accuracy and false alarm rate when the classification is proposed with the number of hidden nodes in the proposed model.

Table I
Comparison of the Proposed Algorithm, the JENN, Constrained JENN and HGNN

Method	**No. of input features**	**No. of hidden nodes**	**Detection Accuracy (%)**	**False Alarm Rate (%)**
Constrained JENN	41	16	87.46	3.45
JENN	15	10	91.51	1.31
HGNN	15	9	92.86	1.14
TIHGNN	13	7	94.72	0.92

From this Table I, 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 used in the classification.

Table I shows the comparison of Constrained JENN, JENN, HGNN and proposed TIHGNN with respect to detection accuracy and false alarm rate when the classification is proposed with the number of hidden nodes in the proposed model.

Table II
Comparison Between the Proposed Algorithm and Other Methods

Method	Detection Rate (%)				False Positive Rate(%)
	DoS	Probe	R2L	U2R	
RWNN	95.53	91.30	60.01	54.70	9.06
BMPNN	96.78	96.05	48.47	38.62	3.12
ENN	97.74	92.20	58.30	52.86	4.83
HGNN	98.28	96.39	60.32	55.17	1.14
TIHGNN	99.21	97.12	61.20	56.87	0.98

From this Table II, 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 used in the classification.

Figure 2 shows the performance comparison between the proposed TIHGNN and the existing model HGNN.

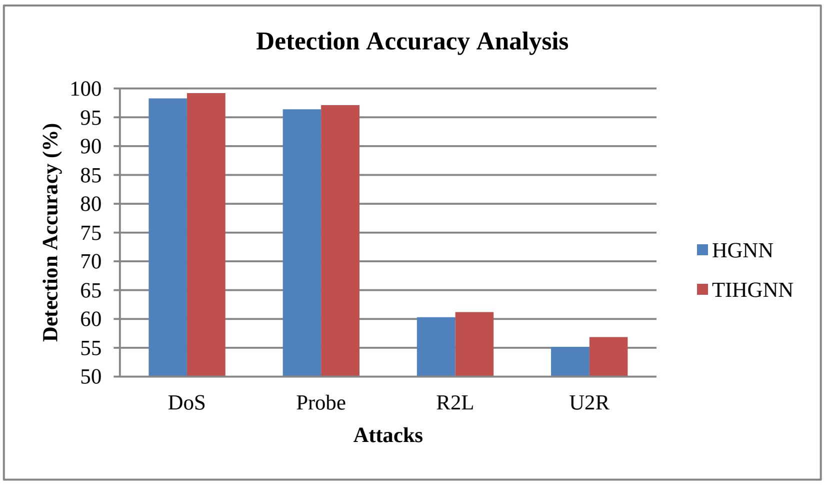

Fig. 2. Performance Analysis

From this figure 2, it is observed that the detection accuracy is high in the proposed model when it is compared with the existing model. This is due to the fact that in the proposed model, detection accuracy is improved by using temporal Constraints.

VI. CONCLUSION AND FUTURE ENHANCEMENT

In this paper, we propose a new Temporal oriented Intelligent Heuristic Genetic Neural Network (TIHGNN) to improve the performance of intrusion detection, in which input features, network structure and connection weights have evolved jointly. The experimental result shows that the proposed TIHGNN accomplishes feature selection, structure optimization and weight adaptation effectively. Through the comparative analysis, it has been seen that the proposed TIHGNN achieves better detection performance and compact structure of neural network.

REFERENCES

[1] Biying Zhang, “A Heuristic Genetic Neural Network for Intrusion Detection”, International IEEE Conference on Internet Computing and Information Services, pp. 510-513, 2011.

[2] Wei Zhang, Shaohua Teng, Haibin Zhu, Hongle Du, Xiaocong Li, “Fuzzy Multi-Class Support Vector Machines for Cooperative Network Intrusion Detection”, Proc. 9th IEEE Int. Conf. on Cognitive Informatics, pp. 811-818, 2010.

[3] 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.

[4] NR Guo, T-HS Li, “Construction of a neuron-fuzzy classification model based on feature-extraction approach”. Expert Syst. Appl 38, 682–691 (2011).

[5] S Ganapathy, P Yogesh, A Kannan, “Intelligent Agent based Intrusion Detection using Enhanced Multiclass SVM”, Computational Intelligence and Neuroscience, Vol. 2012, pp. 1-10, 2012.

[6] Huaguang Zhang, Jinhai Liu, Dazhong Ma and Zhanshan Wang, “Data-Core-Based Fuzzy Min–Max Neural Network for Pattern Classification”, IEEE Transactions on Neural Networks, Vol. 22, No. 12, pp. 2339-2352, 2011.

[7] Oong, T.H. and Isa, A. “Adaptive Evolutionary Artificial Neural Networks for Pattern Classification”, IEEE Transactions on Neural Networks, Vol. 22, No. 11. pp. 1823-1836, 2011.

[8] Gui, H. and Qiao, J. “Adaptive Computation Algorithm for RBF Neural Network”, IEEE Transactions on Neural Networks and Learning Systems, Vol. 23, No. 2. pp. 342 – 347, 2012.

[9] KDD Cup 1999 “Data, Information and Computer Science”, University of California, Irvine.
http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[10] Adetunmbi A.Olusola., Adeola S.Oladele. and Daramola O.Abosede, “Analysis of KDD ’99 Intrusion Detection Dataset for Selection of Relevance Features”, Proceedings of the World Congress on Engineering and Computer Science, Vol. I, 2010.

[11] E.Kesavulu Reddy, Member IAENG, V.Naveen Reddy, P.Govinda Rajulu,” A Study of Intrusion Detection in Data Mining “,Proceedings of the World Congress on Engineering, Vol. III, 2011.

[12] Dewan Md. Farid, Nouria Harbi, Emna Bahri, Mohammad Zahidur Rahman, Chowdhury Mofizur Rahman ,“Attacks Classification in Adaptive Intrusion Detection using Decision Tree”, World Academy of Science, Engineering and Technology, Vol. 63, 2010.