Webology (ISSN: 1735-188X)
Volume 14, Number 2, 2017

Hybrid Feature Selection And Classification Method For Intrusion Detection In Mobile Ad Hoc Networks

M. LALLI [1, *], V PALANISAMY [2]

[1]Department of Computer Science, Bharathidasan University, Trichy.

[2]Department of Computer Applications, Alagappa University, Karaikudi.

Abstract: Mobile Ad-hoc network has turned into an energizing and essential innovation as of late in view of the fast expansion of wireless gadgets. Mobile ad-hoc networks are highly vulnerable against attacks because of the progressively topology changing of network, the absence of centralized point for monitoring and open medium. The different attacks against mobile nodes are Warm hole, Byzantine attack, Packet Dropping, Black hole and flooding so on. It is vital to look new architecture and networks to ensure the wireless networks and the application of mobile computing. Intrusion Detection System devices are reasonable for distinguishing these attacks. It examine the network exercises by method for review information and use examples of surely understood attacks or ordinary profile to recognize potential attacks. In this paper, a hybrid framework to predict the intruding nodes in the Mobile ad-hoc network routing protocols. This framework is composed of three stages. The first stage of the framework is used to reduce the number of features of KDD CUP dataset which consists of six types of Denial of Service attacks by Hybridizing the Information Gain and Relative Reduct then optimal or reduced dataset is obtained. In the second stage of the framework, an Ant Colony Optimization is used for generating the rule structure for the six types of Denail of Service attacks using reduced features. In the second stage, from the optimal dataset, Naïve Bayes classification is used to classify the features into two categories are known attacks and unknown attacks. And a rule structure is generated by Ant Colony Optimization for known as well as unknown attacks. And in the final stage, Artificial Neuro Fuzzy Inference System is used to predict the behavior of the nodes in the Mobile ad-hoc network.

keywords: Mobile Ad Hoc Network, Denial of Service Attacks, Information Gain,

Relative Reduct, Ant Colony Optimization, Naïve Bayes Classification, Artificial Neuro Fuzzy Inference System.

1. Introduction

Wireless Networking is presently the medium of decision for some applications. What's more, modern fabricating systems permit progressively refined usefulness to dwell in gadgets that are ever littler, thus progressively portable. Mobile ad-hoc networks[1] join wireless communication with a high level of node mobility. High mobility and Constrained range wireless communication implies that the nodes must coordinate with each other to give vital networks, with the basic network progressively changing to guarantee needs are consistently met. The dynamic way of the protocols that empower MANET operation implies they are promptly suited to sending in outrageous or unstable conditions. MANETs have therefore turned into an exceptionally prominent research subject and have been proposed for use in numerous regions, for example, rescue operations, strategic operations, ecological observing, meetings, and so forth [2].

MANETs by their extremely nature are more defenseless against attacks than wired systems [2]. The adaptability gave by the open communicate medium and the helpfulness of the cell phones (which have by and large extraordinary asset and computational limits, and run for the most part on battery control) presents new security dangers. As a feature of reasonable risk administration we should have the capacity to recognize these dangers and make suitable move. At times we might have the capacity to plan out specific risks cost-adequately. In different cases we may need to acknowledge that vulnerabilities exist and try to make proper move when we trust somebody is attacking us. Accordingly, intrusion detection is an imperative piece of security for MANETs [3].

2. Problem Statement in the MANET Intrusion Detection System

The vast difference between the fixed network where current intrusion detection research are taking place and the mobile ad-hoc network which is the focus of this paper makes it very difficult to apply intrusion detection techniques developed for one environment to another. The most important difference is perhaps that the latter does not have a fixed infrastructure, and today’s network-based IDSs, which rely on real-time traffic analysis, can no longer function well in the new environment. There are new issues which ought to be considered when another

IDS is being intended for MANETs.

- **Wireless Links [6] [7]**
- **Absence of Central Points [4]**.
- **Limited Resources [8]**
- **Mobility [5]**
- **Cooperativeness [11]**
- **Absence of a Clear Line of Defense and Secure Communication [9][10]**

3. Proposed Framework for Prediction of Risk Severity of the Intruding Node in MANET Intrusion Detection System

The proposed framework is composed of three stages for predicting the intruding nodes in MANET which helps to prevent from different attacks. The following figure 1 represents the proposed framework for predicting the intruding node by classifying them into good, bad, suspiciously good and suspiciously bad.

Stage 1: Hybrid Feature Selection Algorithm (Information Gain and Relative Reduct is hybridized to reduce the number of features in KDD CUP dataset) and the Naïve Bayes classification method is used to further classifies the optimal dataset into known attacks category and unknown attack category.

Stage 2: In this stage, Ant Colony Optimization is used to frame the rule structure for known attacks as well as unknown attacks.

Stage 3: Using the stage 2 result (Rule structure of known and unknown attacks), Artificial Neuro Fuzzy Inference System (ANFIS) predict the severity of the intruding node in the MANET.

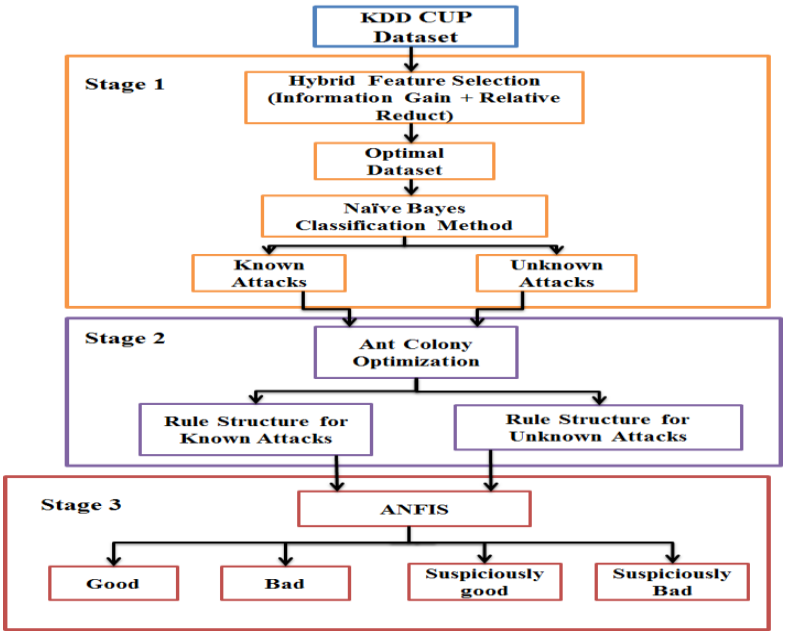

Figure 1: A Novel Framework for Predicting the risk severity of the intruding nodes

4. Data set for Experiment

This dataset contains three sorts of traffics and six sorts of DoS attack around four gigabytes and every traffic record has 41 features names whose values facilitate to recognize the sort classification either as ordinary or attack. It contains a sum of 24 attack sorts that fall into four noteworthy classifications, for example, Test, R2L (Remote to User), U2R (User to Root) and DoS (Denial of Service). DoS attacks are hard to manage on the grounds that they are anything but difficult to dispatch, hard to track furthermore it is difficult to reject the solicitations of the attacker. PoD (Ping of Death), Teardrop, Neptune (Syn Flood), Land, Back and smurf are the six sorts of DoS attacks in KDDCUP 99 [12]. Against the Apache Web Server, the DoS attacks are back sorted, an attacker submits demands with URL's containing numerous front cuts. As the server tries to prepare these solicitations it will back off and gets unable to proceed other request. Back attack wants to realize that solicitations for archives with many numbers of front cuts in the URL ought to be viewed as an attack. In the "smurf " attack, ICMP echo is used by attackers to demand packets coordinated from remote areas to IP broadcast addresses for making a denial of service attack. The Land attack happens when an aggressor sends a spoofed SYN packet in which the source

location is the same as the destination address. Teardrop happens because of IP fragmentation re-assembly code which does not legitimately handle covering IP fragments. This attack by searching for two extraordinarily fragmented IP datagram. The table 1 represents the Different Attacks types in 10% KDDCUP 99 Dataset. And table 2 represents the description of the dataset.

Table 1: Different Attacks types in KDD CUP dataset

5 Main Attack Classes	22 Attack Classes	Samples
Normal		97,277
Denial of Service (DoS)	Back, land, Neptune, pod, Smurf, teardrop	391458
Remote to user	ftp _write, guess _passwd , imap ,multihop, phf, spy, warezclient, warezmaster	1126
User to Root (U2R)	buffer_overflow,perl,load module, rootkit	52
Probing	ipsweep, nmap, portsweep, satan	4107

5. Stage 1: Hybrid Feature Selection Algorithm in Proposed Framework

5.1 Information Gain Feature Selection Method

In this technique, the perceptibility capacity is utilized[13]-[16]. The perceptibility capacity is given as takes after: For a data framework (H,I), s detectability capacity DC is a boolean capacity of m Boolean variables f1,f2… … ..fm comparing to the traits f1,f2..… fm individually, and characterized as takes after: DC(f1,f2… .fm)=Q1 ∧ Q2 ∧… ..Qn where fk € Q. The proposed calculation for the information gain characteristic subset assessment is characterized as given below:

Step 1:Compute perceptibility lattice for the chose dataset. By utilizing S[K,J]={ f € F, where A[K]≠A[J] and B[K]≠B[J]} K,J=1,2,… .n (1)

Where A are condition attribute and B is a decision attribute. This detectability matrix S is symmetric. Where S[c,d]=P[d,c] and P[c.c]=0. Along these lines, it is adequate to consider just the lower triangle or the upper triangle of the matrix.

Step 2:Compute the discernibiliy capacity for the perceptibility matrix S[c,d] by utilizing

DC(c) = ∧ {∧ S[c,d]/c,d € H; S[c,d] ≠0} (2)

Step 3: Select the property, which has a place with the extensive number of conjunctive sets, numbering no less than two, and apply the law of expansion.

Step 4: Repeat steps 1 to 3 until the law of expansion can't be connected for every part.

Step 5:Substitute all unequivocally proportional classes for their relating properties.

Step 6:Calculate the Information gain for the improved perceptibility capacity contained qualities by utilizing Gain(Ij) = F(Pj) - F(Ij) (3)

Where $F(I) = \sum_{k=1}^{n} P_K log_2 P_K$ (4)

$$\sum_{k=1}^{n} P_k log_2 P_k = -\frac{p1}{p} log_2 \frac{p1}{p} - \frac{p2}{p} log_2 \frac{p2}{p} - \frac{pn}{p} log_2 \frac{pn}{p} \quad (5)$$

Where Pk is the proportion of contingent quality P in dataset. At the point when Ij has | Ij | sorts of property estimations and condition characteristic Pk segments set P utilizing trait Ij, the estimation of data F(Gi) is characterized as

$$F(I) = \sum_{j=1}^{Ij} Wj * F(Ij) \quad (6)$$

Step 7: Choose the most elevated Gain esteem and add it to the lessening set, and expel the trait from the perceptibility capacity. Goto step 6 until the perceptibility capacity achieves invalid set.

5.2 Relative Reduct

Relative Reduct Algorithm is the most understood calculation for feature selection utilizing Rough sets [14][15]. This is an incremental methodology; where it begins with a void set and in every stride a feature is added to the Reduct, in such way that dependency quantifies increments. The methodology stops when the dependency measure of the arrangement of elements being considered is equivalent to the dependency measure utilizing all the conditional features. The calculation endeavors to figure a reduct without comprehensively producing every single conceivable subset [15]. Its pseudo code calculation is given underneath:

Input: Original Dataset,

D the set of all conditional features; S- the set of decision features, a reduct is defined as Q subset.

Input: Original Dataset

Begin

Initialize Q as Empty set (is represented by {})
R ← Q
∀χ ∈ (D - Q)
if γ_(Q ∪{χ}) (S)> γq (S)
R ← Q ∪ { χ }
Q ← R
Until γq (S) = γd (S)
Return Q
End
Output: A Reduct Dataset

The Reduct Relative algorithm endeavors to ascertain a reduct without completely creating every single possible subset. It begins off with a vacant set and includes turn, each one in turn, those features that outcome in the best increment in the rough set dependency metric, until this delivers its most extreme conceivable quality for the dataset.

5.3 Hybrid Feature Selection

The following gives brief explanation about the hybrid feature selection which integrates the relative reduct and information gain for reducing the feature space size.

Input: S (A0, A1,... Am-1) // A training dataset with M features
R0 // a subset from which to start the search

Algorithm:

Begin
Initialize:$R_{best} = R_0$;
d_0 = card (R_0) // Calculate the cardinality of R_0
$\theta_{best} = eval\ (R_0, S, I)$; // evaluate S0 by an independent measure I
$\gamma_{best} = eval\ (R_0, S, Q)$; // Evaluate S_0 by a Relative Reduct algorithm Q
for d = d_0+1 to M begin
for j =0 to M-d begin
R= R_{best}∪ {Ai};//For evaluation, a subset with cardinality c is generated
θ = eval (R, S, I); //The current subset is evaluated by Independent measure I
if (Θ is better than Θ_{best})

$\Theta_{best} = \Theta$;

$R'_{best} = R$;

end;

γ = eval ($R'_{best}, S, F(G)$) //evaluate R'_{best} by Information Gain Algorithm

if (γ is better than γ_{best})

$R_{best} = R'_{best}$;

$\gamma_{best} = \gamma$;

Else;

Break and return R_{best};

End;

Return R_{best}

Output: Reduct Dataset

In the pseudo code, S represents the Original dataset with list of attributes; R_0 is used to start the search by using sequential forward search method. d_0 holds the result of the cardinality of R_0. Θ_{best} is the evaluation of the cardinality of the dataset by means of independent measure. γ_{best} is the evaluation of cardinality in the dataset by using Relative Reduct algorithm. Up to M-d γ attributes, the cardinality is calculated by R from R_{best}. is the evaluation of the R_{best} by Information Gain algorithm.

5.4 Naïve Bayes Classification

In this phase, the outcome of the above proposed hybrid feature selection algorithm is given as an input to the Naïve Bayes classification to classify the attacks as Known attacks and unknown attacks. Naïve Bayesian classifier [16] is a simple classification scheme, which estimates the class- conditional probability by assuming that the attributes are conditionally independent, given the class label.

Naive Bayes is a strategy for assessing probabilities of individual variable qualities, given a class, from preparing information and to then permit the utilization of these probabilities to order new elements, which is a term in Bayesian insights managing a straightforward probabilistic classifier taking into account applying Bayes' hypothesis (from Bayesian measurements) with strong (guileless) autonomy assumptions. In basic terms, a strong Bayes classifier expect

that the nearness (or nonappearance) of a specific feature of a class is disconnected to the nearness (or nonattendance) of some other element. The Naive Bayesian classifier, fills in as taking after inference [16]:

Step 1: Let T be a training set of tuples and their related class names. Each tuple is spoken to by a m-dimensional attribute vector, A = (a1, a2, … .. , am), m estimations made on the tuple from m properties, individually, X1, X2, … , Xm.

Step 2: Suppose that there are n classes D1, D2, … . , and Dn. Given a tuple, A, the classifier will anticipate that A has a place with the class having the most noteworthy back likelihood, adapted on A. That is, the guileless Bayesian classifier predicts that tuple A has a place with the class Tj if and just if

$$P\left(\left(D_j|A\right) > P\left(\left(D_k|A\right)\right)\right) \text{ for } 1 \leq k \leq n, k \neq j \quad (7)$$

The boost P(Dj|A). The class Dj for which P(Dk|A) is amplified is known as the most extreme posterior hypothesis. By Bayes' hypothesis (Next condition)

$$P\left(D_j|A\right) = \frac{P\left(A|D_j\right)P(D_j)}{P(A)} \quad (8)$$

Step 3: Since P(A) is consistent for all classes, just (P(Dj|A) = P(A |Dj)P(Dj)) should be amplified.

Step 4: Based on the supposition is that properties are restrictively free (i.e., no reliance connection between attributes), the registering of P(A|Dj) utilizing the accompanying condition:

$$P\left(A|D_j\right) = \prod_{i=1}^{m} P\left(a_i|D_j\right) \quad (9)$$

Diminishes the calculation cost by Equation (P(Dj|A) = P(A |Dj)P(Dj), just numbers the class appropriation. On the off chance that Xi is unmitigated, P(Ai|Dj) is the no. of tuples in Dj having esteem Ai for Xi separated by |Dj, T| no. of tuples of Dj in T. Also, if Xi is persistent esteemed, P(Ai|Dj) is typically processed in view of Gaussian circulation with a mean μ and standard deviation σ and P(Ai|Dj) is:

$$g\left(x, \mu, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (10)$$

$$P\left(D_j|A\right) = g\left(a_i, \mu_{Dj}, \sigma_{Dj}\right) \quad (11)$$

Where μ is the mean and σ is the difference. On the off chance that a property estimation doesn't happen with each class esteem, the likelihood will be zero, and

a posteriori likelihood will likewise be zero.

6. Stage 2: Generation of Rule Structure by Ant Colony Optimization

The ACO is a probabilistic method for taking care of computational issues which can be decreased to discovering great paths through graphs taking into account the systems of genuine ants. It was at first proposed in 1992 by Colorni, Dorigo and Maniezzo [17][18]. In ACO, each artificial ant is considered as a straightforward operator, speaking with different ants just in a roundabout way and by affecting changes to a typical situation.

Algorithm for Ant Colony Optimization

Step 1: The Pheromone Trail is initialized
Step 2: While halting criteria = not met do
Step 3: For all ants do
Step 4: Randomly deposit ant
Step 5: While solution=incomplete do
Step 6: According to Pheromone trail, select next component randomly
Step 7: end while
Step 8: end for
Step 9: Pheromone trail is updated
Step 10: end while

In the year 1992 for PhD theory of Marco Dorigo [18], the principal algorithm was intending to hunt down an ideal path in a graph, considering the conduct of ants looking for a path between their settlement and a source of nourishment. The first thought has ensuing to improve to tackle a more broad class of numerical issues, and accordingly, a small number of issues have developed, on the different parts of the conducting of ants is drawn. The primary elements are fast search speed, high-precision arrangement, union to worldwide ideal and eager heuristic hunt.

Probabilistic Transition Rule: The random proportional rule is also called as the Probabilistic Transition Rule

$$Q_{jk} = \frac{[\tau\eta_{jk}]^{\beta}.[\eta_{jk}(s)]^{\alpha}}{\sum_{j=1}^{c}\sum_{k=1}^{d_j}(\tau_{jk}.\eta_{jk}(s))} \forall j \in J \quad (12)$$

where: Q_{jk} is the probability rule antecedent, τ_{ij} is the value of heuristic, at iterations s, the pheromone amount is $\eta_{ij}(s)$, the total number of attributes is given by a, d_j is the number of domain values of the j-th attribute, J are the attributes not yet used by the ant and (c and d) are two adjustable parameters is used to control the relative weights of the values of pheromone and heuristic respectively.

Quality Computation: Quality computation includes that the quality of a rule in Ant-Miner is computed according to as given in

$$DQ = \frac{TrN}{(TrN + FaP)} X \frac{TrP}{(TrP + FaN)}$$

True Positive – TrP; False Negative – FaN; True Negative – TrN;
False Positive - FaP

As it is mentioned, between the discovered ones, the best rule is selected by using the quality. Besides, Ant-Miner uses the quality as a factor for pheromone updating.

Pheromone Updating: Pheromone updating includes that, in each iteration s, the pheromone will be increased for all the terms including in the constructed rule as given in

$$\eta_{ij}(s+1) = \eta_{ij}(s) + \eta_{ij}(s) * DQ \ \forall j,k \ \in P$$

where P is the set of all the terms included in the rule. Furthermore, the pheromone should be decreased $\eta_{ij}(s)$ for every not in the antecedent part of the rule. By normalizing the pheromone this objective would be satisfied.

7. Stage 3: Prediction of Intruding Node severity using Artificial Neuro Fuzzy Inference System

In this phase, the characteristic of the node that is used for the routing in the MANET is considered. Then the properties of the node is compared with phase 1 rule structure of known attacks and phase 2 rule structure of unknown attacks. Neuro Fuzzy System (ANFIS- Artificial Neural Fuzzy Inference System) to further classify the node as good, bad, suspiciously good and suspiciously bad which would be used for routing. This phase is used to know whether the node is considered for routing or not. The figure 3 represents the framework for the prediction of Node Behavior using Artificial Neural Fuzzy Inference System

(ANFIS) or Neuro Fuzzy System.

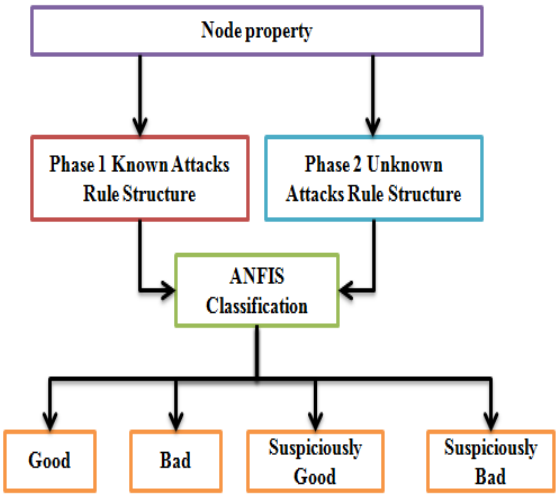

Figure 2: Framework for the prediction of Node Behavior using ANFIS

This is the combination of Artificial Neural Network and Fuzzy logic to predict the node behaviors in the MANET routing. The node is considered only routing if the node is predicted as good one. The procedure of a fuzzy framework has three stages. These strides are Rule Evaluation, Fuzzification and Defuzzification. In the step of fuzzification, in the fuzzy sets, the information input values are changed into degrees of membership. In the standard assessment step, each fuzzy guideline is allotted with a quality worth. In the fuzzy guideline of arrangements of forerunner part, the quality is controlled by the degrees of memberships of the crisp input values. The defuzzification stage transposes the fuzzy yields into input values. The fuzzy guideline calculation is produced for making the preparation dataset for the FNNM (Fuzzy Neural Network Machine). The SQL inquiries are created utilizing the "RandAndOr" capacity for short-posting the unmistakable qualities contained in every field. These qualities symbolize the attributes of irregularity in the interruption information and typicality from the ordinary information. The tenet creation produced an intelligent grouping which contains the "and" and "or" legitimate administrators and effect the choice of irregularity or typicality are spoken to as far as weights relegated. Neuro Fuzzy System is developed utilizing the accompanying calculation.

Pseudo Code: Algorithm of Fuzzy Rule

Input: Optimal Data Set
Start
S=selected attribute
K=operation subset
A=next component from the accessible information
K=item[j]
For j=1 to m-1
A=DataFIeld [j+1]
K=K union A in the field the special thing is selected
End for
Store K
End
Initialize Increment to 1
Initialize Weight of Find Record to 0
Introduce Quct to 1
WHILE Increment < MJ
FOR every worth FuL
Record [FuL] = rand() mod Nfl
END FOR
FOR every worth JF
Qust = sql select proclamation where each
Field[JF] = Index[JF] + " + RandAndOr();
END FOR
TotalFuzzyR = ExecuteQuery(Qust)
If TotalFuzzyR is non zero THEN
Wht[Quct] = TotalR/TotalFuzzyR
1 is added to Quct
ENDIF
1 added to Increment
ENDWHILE
Save Wht
Save Qust
Output: Node Behavior: Good, Bad, Suspiciously Good and Suspiciously Bad.

8. Experimental Result and Discussion

The proposed computational hybrid Intrusion Detection model for identifying the intruded nodes was implemented in MATLAB. Amid the assessment, 10 percent named information of KDDCUP 99 was utilized for preparing the proposed hybrid intrusion detection model for identifying the node behavior for routing in MANET. With the payload of size M, the principle datagram would be 0 in counterbalance section, with the NP (Next Point) bit on (the information substance of the packet is unimportant). The second datagram is the last section (NP = 0), with a positive balance more prominent than M and with a payload of size not as much as M. Neptune attack depicts that every TCP connection of half-open made to a machine causes the "tcpd" server to add a record to the information structure that stores data about every single pending connection. This information structure is of limited size, and it can be made to flood by purposefully making excessively numerous incompletely open communications. Neptune attack can be recognized from ordinary network traffic by searching for various synchronous SYN packets bound for a specific machine that are originating from an inaccessible host. A host-based intrusion detection framework can screen the extent of the tcpd connection information structure and alarm a client on the off chance that this information structure nears its size point of confinement. Ping of Death attack has been accounted for when the framework respond in a flighty manner while getting larger than average IP packets. Conceivable responses incorporate smashing, solidifying and rebooting. Ping of Death can be distinguished by noticing the span of all ICMP packets and flagging those that are longer than 64000 bytes.

8.1 Stage 1: Experimental Result and Discussion

To approve the outcomes got from the hybrid intrusion detection calculation, the accompanying parameters Kappa Statistic, Performance, False Positive Rate (FPR), True Positive Rate (TPR), Relative Absolute Error (RAE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Root Relative Absolute Error (RRAE), Mean Squared Error (MSE), Receiver Operating Characteristic Curve (ROC), Confusion Matrix, Precision, Recall are considered. The average square difference between the outputs and targets is Mean Squared Error. If zero it means no error whereas lower values are better. The correlation between targets and output is measured by value called Regression R. The random relationship is indicated by 0 whereas close relationship is given by 1. The TPR against FPR is considered to plot the ROC curve. The obtained result is considered as good

only when the ROC value areas are nearer to the value of 0.80 to 0.90. The predictive model performance is estimated by using Cross Validation technique. In the 10 fold cross validation, the data sets are divided into 10 sets in which 9 data sets are used for training and 1 is used for testing.

Table 3: Stage1 ResultNumber of Feature Selected from original Dataset using Information Gain, Relative Reduct and Proposed Hybrid Feature Selection (IG+RR)

Feature Index	Information Gain Technique	Relative Reduct Rough Set Technique	Proposed Feature Selection (IG + Relative Reduct)
1	Duration	Duration	Protocol_type
2	protocol_type	Service	Service
3	Service	protocol_type	Src_Byte
4	Flag	Flag	Wrong_fragement
5	src_bytes	src_bytes	Dst_host_count
6	dst_bytes	dst_bytes	Flag
7	Count	Same_srv_rate	Same_srv_rate
8	srv_count	Srv_Count	Srv_Count
9	serror_rate	Dst_host_diff_Serv_rate	Dst_host_diff_Serv_rate
10	srv_serror_rate	Dst_host_srv_error_rate	Dst_host_srv_error_rate
11	same_srv_rate	Wrong_fragement	Duration
12	diff_srv_rate	dst_host_same_srv_rate	Dst_byte
13	srv_diff_host_rate	dst_host_diff_srv_rate	
14	dst_host_count	dst_host_srv_serror_rate	
15	dst_host_srv_count	srv_diff_host_rate	
16	dst_host_same_srv_rate	dst_host_rerror_rate	

17	dst_host_diff_srv_rate		
18	dst_host_srv_serror_rate		
19	dst_host_rerror_rate		
20	dst_host_srv_rerror_rate		

Table 4: represents the comparison of feature selection using Information Gain, Relative Reduct and Proposed Hybrid Feature Selection algorithm from the given original dataset. This dataset contains six types of Denial of Service Attacks. For these types of attacks, the proposed Hybrid Feature Selection method selects least number of records than the other techniques like Information Gain, Relative Reduct.

Table 4: Comparison of Feature Selection Result using Original dataset, Information Gain, Relative Reduct and Proposed Hybrid Feature Selection Method

Attack Type	Original Dataset	Information Gain Technique	Relative Reduct Rough Set Technique	Proposed Feature Selection (IG + Relative Reduct)
Normal	97, 277	97,277	97,277	97,277
Tear Drop	979	805	762	698
Back	2203	1985	1754	1685
Smurf	2,80,790	60451	10789	1024
Pod	264	189	157	123
Neptune	107201	50421	15478	845
Land	21	19	17	15

The Table 5 and Table 6 represent the performance comparison of Proposed Hybrid Feature selection method, Hybrid Classification Model with existing Artificial Neural Network. From the table 5, the intruded nodes detection rate is higher for Proposed Hybrid Feature Selection method than other two techniques. The error rate and execution time is decreased for the proposed method. In the Table 6, it is proved that the error rates like MAE, RMSE, RAE, RRAE, FPR are reduced. TPR, Kappa Statistic, Precision, Recall, ROC, Classification accuracy are increased by using Proposed Hybrid Classification methodology than Proposed Hybrid Feature Selection with Artificial Neural Network.

Table 5: Performance Comparison of Relative Reduct, Information Gain and Proposed Hybrid Feature Selection Method

Performance Metrics	Quick Reduct	Information Gain	Proposed Hybrid Feature Selection Method(RR+IG)
Deduction(in %)	86%	85%	92%
Execution Time	3.54 mins	2.96 mins	0.98 mins
Error Rate	88%	85%	70%

Table 6: Performance Comparison of Proposed Hybrid Classification Model and Artificial Neural Network Model

Performance Metrics	Hybrid (RR + IG) + NaïveBayes Classification	Hybrid (IG+RR) +Artificial Neural Network
Correctly Classified Instance	82.61	79.74
Kappa Statistic	0.66	0.45
Mean Absolute Error	0.22	0.26

Root Mean Squared Error	0.35	0.44
Relative Absolute Error	46.25	52.57
Root Relative Absolute Error	78.64	87.67
True Positive Rate	0.81	0.69
False Positive Rate	0.21	0.32
Precision	0.81	0.69
Recall	0.81	0.69
Receiver Operating Characteristic Curve	0.84	0.73

8.2: Stage 2 and Stage 3: Experimental Result and Discussion

The Table 7 represents the generation of rule structure for six types of known attacks using phase 1 proposed framework. Using the feature selected from Hybrid Feature Selection method, the following rules are constructed using Ant Colony Optimization based Classifier. These rule structure are used to classify the nodes into the intruded category if it behaves as malicious. And table 8 represents the types of unknown attacks using the rule structure for six types of known attacks.

Table 7: Generation of Rule Structure for six types of Known Attacks by Proposed Framework Phase 2 using Ant Colony Optimization

S.No	Attack Description	Attack Type
1	Protocol=UDP, Service=SF, wrong fragment=3, dst_host_count=255 and If (source_bytes> 265616) and(source_bytes<= 283618) Then Warezmaster Attack	Tear drop
2	Protocol=tcp, service=http, flag=SF or RSTFR, src_byte=54540, dst_byte=7300	Back

	or 8314, same_srv_rate=1, srv_count>=5	
3	Protocol=ICMP, service=ecr_i, src_byte=1032, flag=SF, host_count=255 If (Duration <3) and(dst_byte=125016) Then Buffer overflow	smurf
4	Protocol=ICMP, service=ecr_i, flag=SF, src_byte=1480, wrong_fragment=1, dst_host_count=255, dst_host_diff_srv_rate =0.02. If(duration<10seconds) of an FTP connection /session, there are many Hot indicators (hot > 20) being set by a logged user then it is highly likely that is being executed.	Pod
5	Protocol=tcp, service=private or ctf, flag=SO or SF, serror_rate=1 if {the connection has following information: source IP address 124.12.5.18; destination IP ad-dress:130.18.206.55; destination port number: 21; connection time: 10.1 seconds } Then {stop the connection	Neptune
6	Flag=SO, land=1, srv_count=2, dst_host_srv_error_rate>=0.17 If (Duration 0 to 25) and (protocol_ type = tcp and UDP) and (service=ftp OR private OR other domain)	Land

Table 8: Generation of Rule Structure for unknown attacks by Ant Colony Optimization using above Rule structure

S.No	Attack Description	Attack Type
1	Dst_host_count = 255, wrong fragment = 2, Protocol_type = TCP, Service = SF and if (source_bytes> 265616) and(source_bytes<=	Tear drop like attack

	283618) then Warezmaster like attack	
2	Flag = RSTFR, protocol_type = UDP, src_byte <54540, dst_byte<8314, same_srv_rate = 0, srv_count<5	Back like attack
3	Protocol=ICMP, service=ecr_i, src_byte=255, flag= RSTFR, host_count=255 If (Duration = 0) and(dst_byte=1032) Then Buffer overflow like attack	Smurflike attack
4	Protocol=ICMP, service=ecr_i, flag=RSTFR, src_byte=1032, wrong_fragment=1, dst_host_count=1032, dst_host_diff_srv_rate =0.07. If(duration<30seconds) of an FTP connection /session, there are many Hot indicators (hot >50) being set by a logged user then it is highly likely that is being executed.	Pod like attack
5	Protocol=UDP, service=private or ctf, flag=SF, serror_rate=3 if {the connection has following information: source IP address 124.12.5.18; destination IP ad-dress:130.18.206.55; destination port number: 51; connection time: 30.1 seconds } Then {stop the connection	Neptune like attack
6	Flag=SO, land=0, srv_count=5, dst_host_srv_error_rate<=0.17 If (Duration 0 to 50) and (protocol_ type = TCP and UDP) and (service=ftp OR private OR other domain)	Land like attacks

Table 9: Output Representation of the Node Behavior using ANFIS

Input Range	Fuzzy Value
Node Behavior	<1 – Good
	1.1-2 - Bad
	1.8-3 – Suspiciously Good
	>3 –Suspiciously Bad

Table 9 represents the output representation of the node behavior using ANFIS and Figure 3 gives the rule view of ANFIS in the node severity prediction of the MANET.

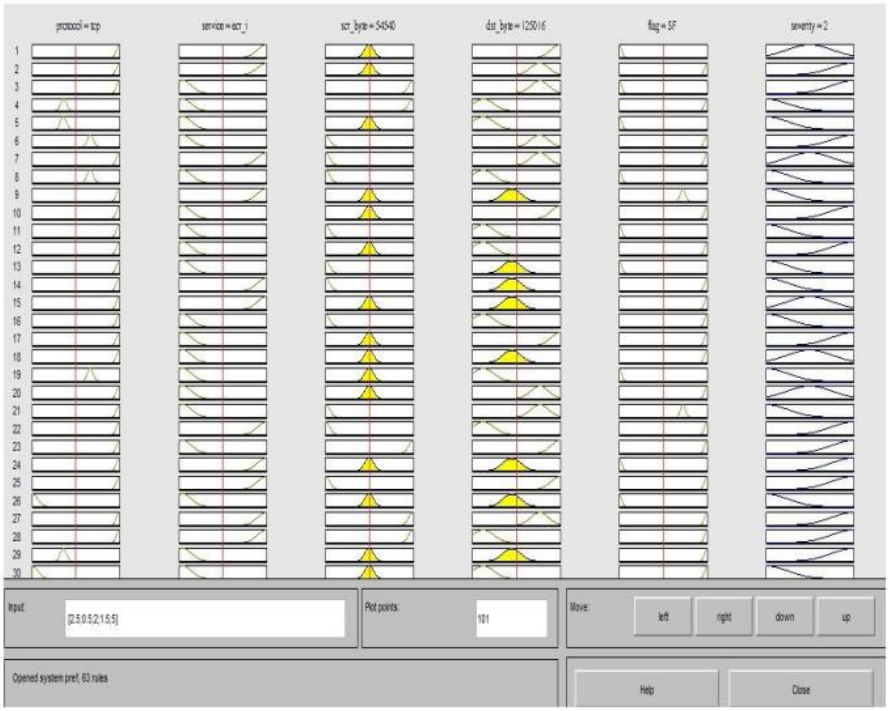

Figure 3: Rule View of ANFIS in Severity Prediction of Suspicious Node

9. Conclusion

Through this research, a new computational hybrid Feature Selection and Classification Model, Framework for the Node Behavior prediction was proposed by hybridizing the Relative Reduct Rough Set and Information Gain for removing the features from the duplications. The classifier based on Ant Colony Optimization is utilized to generate the structure of rules for the six types of known attacks existing in the given dataset. At that point Naïve Bayes Classification is utilized to arrange the given set of attributes as Attacks of Unknown and Known ones. Finally, the last phase gives the prediction framework for the node behavior in the MANET. Using this framework, the node can be predicted as good, bad, suspiciously good and suspiciously bad for using that node in the routing of MANET. Using the proposed three phases, before the node intrusion, the nodes in the MANET can be identified, and the intrusion of the node

can be prevented. This proposed Hybrid Classification Model are adequate interoperability and high reliability and practically identical with a few surely understood calculations like Artificial Neural Network. Results on the given dataset demonstrate that the proposed hybrid classification model would be fit for arranging the nodes by different sorts of attacks with higher exactness. The consequences of hybrid classification model are better than Artificial Neural Network Classification.

References

[1] A. Anand, R. Rani, H. Aggarwal, "A Security Model based on Reputation and Collaboration through Route-Request in Mobile Ad Hoc Networks", KSII Transactions on Internet and Information Systems, Nov2015, Vol. 9 Issue 11, p4701-4719.

[2] Darshana Sorathiya, Haresh Rathod, "Algorithm to Detect and Recover Wormhole Attack in MANETs", International Journal of Computer Applications, Volume 124, no: 14, August 2015,pp.7-11.

[3] Batika Rai, Prof. Anurag Jain, "The APT Identification and Blocking through IDS in MANET", International Journal of Current Trends in Engineering and Technology, Volume 2, Issue 2, April 2016, pp. 246-251.

[4] P. Visalakshi, S. Anjugam, "Security Issues and Vulnerabilities in Mobile Ad Hoc Networks", International Journal of Computational Engineering Research, National Conference on Architecture, Software System and Green Computing, pp.189-194.

[5] Sunil Kumar Singh, Rajesh Duvvuru, Jyoti Prakash Singh, "TCP and UDP based performance evaluation of proactive and reactive routing protocols using mobility models in MANETs", International Journal of Information and Communication Technology, Volume 7, Issue 6, pp. 632-644.

[6] Kavitha Rani, Aarti, "Elimination of Gray Hole Attack using Directional Based Credit Technique in MANET", International Journal of Engineering Science and Computing, pp. 8101-8105, June 2016.

[7] Sunil Kumar, Kamlesh Dutta, "Intrusion Detection in Mobile Ad Hoc Networks: Techniques, Systems, and Future Challenges", Security and Communication Networks, Volume 9, Issue 14 25 September 2016, pp.2484–2556.

[8] Sushma Kushwaha, Prof.Vijay Lokhande, "Security in Wireless Mobile Ad Hoc Network Nodes using Novel Intrusion Detection System", International Journal of Engineering Science and Computing, April 2016, pp.3352-3356.

[9] Gaurav Somani, Manoj Singh Gaur, Dheeraj Sanghhi, Mauro Conti and Rajkumar Buyya, "DDoS Attacks in Cloud Computing: Issues, Taxonomy, and Future Directions", ACM Computing Surveys, December 2015.

[10]D Rajagopa and Thilakavalli K, "Monitoring Internet Access along with Usage of Bandwidth using Intrusion Detection System", International Journal of Sensor Networks and Data Communications, 2015.

[11]Mahadev A. Gawas, Lucy J. Gudino, K. R. Anupama, "Cross Layered Adaptive Cooperative Routing Mode in Mobile Ad Hoc Network", Communications (APCC), 2016 22nd Asia-Pacific Conference, August 2016.

[12]Shashikant Upadhyay, Rajini Ranjan Singh, "Comparative Analysis based Classification of KDD'99 Intrusion Dataset", International Journal of Computer Science and Information Security, Volume 13, Number 3, March 2015, pp.14-20.

[13]Danny Roobaert, Grigoris Karakoulas and Nitesh V. Chawla, "Information G21a in, Correlation and Support Vector Machines", Springer, pp -463-470, 2006.

[14]Xiuyi Jia, Lin Shang, Bing Zhou, Yiyu Yao, "Generalized Attribute Reduct in Rough Set Theory", Three-way Decisions and Granular Computing, Knowledge Based System- Elsevier, Volume 91, January 2016, pp. 204-218.

[15]Jun Wang, Jiaxu Peng, Ou Liu, "A Classification Approach for Less Popular Webpages based on Latent Semantic Analysis and Rough Set Model", Expert Systems with Applications - Elsevier, Volume 42, Issue 1, January 2015, pp. 642-648.

[16]Rong Zhen, Yongxing Jin, Qinyou Hu, Zheping Shao, Nikitas Nikitakos, “Maritime Anomaly Detection within Coastal Waters based on Vessel Trajectory Clustering and Naive Bayes Classifier”, The Journal of Navigation, 2017.

[17]M. Gilli, (2004) "An Introduction to Optimization Heuristics", Department of Econometrics, University of Geneva and FAME www.unige.ch/ses/metri/gilli, Seminar University of Cyprus.

[18]Hevin Rajesh, Paramasivan, “Cluster based Secure Authentication Technique using Ant Colony Optimization in Wireless Sensor Networks”, Journal of Intelligent and Fuzzy Systems, Volume 31, Number 1, pp. 423-432, 2016.