International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 12 (2015) pp. 30413-30430
© Research India Publications
http://www.ripublication.com

# Evolutionary Algorithms For Feature Selection and Classification In Intrusion Detection System

**[1]R.Sridevi, [2] G.Jagajothi**
*[1]Shri Angalamman College of Engineering and Technology, Department of Information Technology, Tiruchirappalli, Tamilnadu, India*
*[2]Periyar Maniammai University, Department of Information Technology, Tanjore, Tamilnadu, India*
*E-mail: [1]sridevi.odm@gmail.com*

## Abstract

Intrusion Detection Systems (IDS) detect attacks against computer systems and networks, or against information systems. It is hard to ensure provably secure information systems, maintaining them in a secure state during their life and use. Sometimes legacy/operational constraints prevent definition of a totally secure information system. Hence, IDS have to monitor usage of such systems to detect insecure states. This study focuses on IDS for classification of normal and abnormal behavior of network traffic data and automatic intrusion rules generation. In the suggested method Genetic search methods are used for feature selection with Artificial Immune System (AIS) being used for system classification.

**Key Words:** NSL KDD Dataset, Artificial Immune System (AIS), Genetic algorithm (GA), Correlation-based Feature Selection (CFS)

## Introduction

Intrusion Detection System (IDS) is the detecting of inappropriate or suspicious activity against computers or networks systems. It is hard to maintain computer systems and networks devices up to date today; numerous breaches are seen daily. IDS monitors systems usage and detects insecure states apparition which can either be internal users attempt to abuse privileges or outside users (attackers) trying to exploit security vulnerabilities [1].

A way to detect intrusions is by using operating system generated audit data. An audit trail records system activities, logged to a file, in a chronological order. As most activities are logged on a system, manual logs inspection would permit intrusion detection. IDSs automate work of wading through an audit data jungle. Audit trails are useful as they establish attacker's guilt, and are usually the only way to detect unauthorized, subversive user activity.

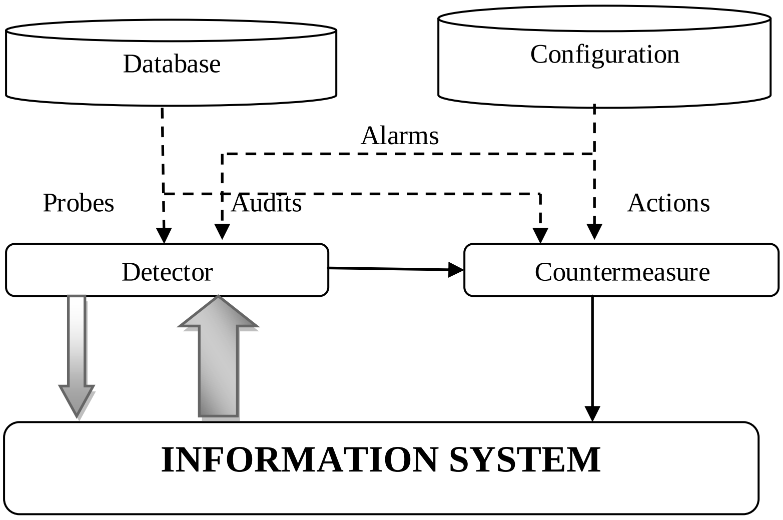

**Figure 1:** Typical Intrusion Detection System

Usually, even after an attack, audit data should be analyzed to determine damage and to help to track down attackers, and also help in steps to prevent this in future. IDS can also analyze audit data for making it valuable as real-time and post-mortem analysis tool.

Intrusions are split into 6 types [2]

1. Attempted break-ins, detected by typical profit behavior or security constraints violations.
2. Masquerade attacks, detected by profiles atypical behavior or security constraints violations.
3. Security control system penetration detected by monitoring specific activity patterns.
4. Leakage detected by atypical system resources use.
5. Denial of service detected by atypical system resources use.
6. Malicious use detected by atypical behavior profiles, security constraints violations or special privileges use.

Anomaly detection is finding patterns in data not conforming to anticipated behavior which are called outliers, anomalies, exceptions, discordant observations aberrations, peculiarities, surprises or contaminants in various application domains. Of these, anomalies and outliers are used commonly in anomaly detection interchangeably. Anomaly detection finds use in various applications like fraud detection in credit cards, insurance/health care, cyber-security intrusion detection, safety critical systems fault detection and military surveillance of enemy activities.

Anomaly detection's importance is due to anomalies in data translating into significant (often critical) actionable information in various application domains. For example, a computer network's anomalous traffic pattern could mean a hacked computer is forwarding sensitive data to unauthorized destination [3]. Anomaly detection is dependent on two assumptions: First, anomaly systems are different form non-intrusive activities at some observation level. Second, intrusions are accompanied by manifestations sufficiently unusual to permit detection [4].

In anomaly based IDS, an established performance baseline reflects what normal network activity is. Network traffic is sampled or compared to this to determine whether the activity is normal. Even with different IDS types available, there is no guarantee that intruders will not tamper systems. A dominant weakness is the time lag in updating and speed reduction due to traffic monitoring within systems. The high number of false alarms is also an IDS concern [5].

Within network security, anomaly detection is one of two approaches used for ID technology with misuse-based (called signature-based) techniques. Despite proved benefits, a major limitation of current misuse-based mechanisms is their inability to detect patterns not previously found in attack signature library. Fig. 2 illustrates best detection results for various intrusion types used in 1999 DARPA ID Evaluation at MIT Lincoln Labs. In Denial-of-Service (DOS) and Remote-to-Local (R2L) attacks, detection rate with new attacks is poor. As in fields like virus/worm detection, reliance on a system whose effectiveness is rooted in patterns knowledge is not bad, though it is not a good operation with unknown activities. Alternatively, anomaly detection approach is conceived as a powerful method due to its theoretical potential to address novel or unforeseen attacks.

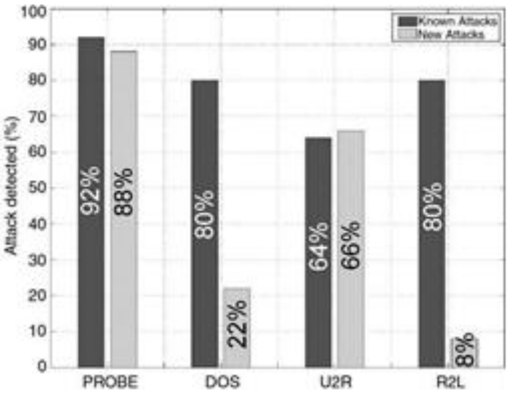

**Figure 2:** Detection results of the top three IDS for the four different intrusion classes used in the 1999 DARPA/MIT Lincoln Labs evaluation

Anomaly detection methods as an early warning system component for network security are the focus of this article. The problem through the presentation of various aspects is presented in section 2: theoretical considerations, general architecture of anomaly detection systems, and historical background [6].

GA's [7] employ metaphors from biology and genetics for iterative population evolvement of initial individuals to population of high quality individuals, with each individual represents a problem solution to be solved, composed of a fixed genes number. Number of possible values for each gene is called cardinality of that gene. Each individual is a chromosome and a set of chromosomes forms a population.

GA evolves new rules for IDS, using which normal network traffic or audit data is differentiated from abnormal traffic/data. Rules in GA rule set are of type if-then. Following is GA general syntax for rule:

If {condition} then {act}

Condition refers to data needing verification and rule in rule set is action to be performed when condition is true. A condition checks port numbers of network protocols, protocols used, connection duration, source and destination's IP address etc. while act is the action to be implemented in a true condition like sending alert message and creating log messages.

The benefits of using GA for intrusion detection are:

a) GA's are intrinsically parallel. Due to multiple offspring, they explore solution space in multiple directions simultaneously.
b) Parallelism permits GA to evaluate many schemas at once making them suited to solving problems where potential solution space is huge.
c) GA based systems are re-trained easily improving its possibility to add new rules and evolve IDS.

This study focuses on IDS for efficient classification of network traffic data's normal and abnormal behaviour and generation of automatic intrusion rules. Genetic search methods are used in the proposed method for feature selection, while Artificial Immune System (AIS) is used for system classification.

## Literature Survey

An Evolution Induced Secondary Immunity and AIS Based IDS was proposed by Dal, et al., [8] describing a technique of applying artificial immune system with GA to develop an IDS. Far from developing primary immune response as most related works do, it attempts to evolve a primary immune response to a secondary immune response using memory cells concept present in natural immune systems.

A hybrid of rough set theory and Artificial Immune Recognition System to reduce false alarm rate in IDS was proposed by Sabri, et al., [9]. This study was improved by rough set theory integration with Artificial Immune Recognition System 1 (AIRS1) algorithm (Rough-AIRS1) to categorize DoS samples. Results from experiments revealed that Rough AIRS1 had lower false alarm rate compared to single AIRS but a little higher than J48 but this hybrid technique's accuracy was lower when compared to others.

Design of a new distributed model for IDS based on AIS that proposed a distributed multi-layered framework to enhance IDS detection performance and efficiency was proposed by Hosseinpour, et al., [10]. In this GA enhanced secondary immune response the aim was to reduce detection time for every connection by distributing detectors to each host.

An analysis on AIS as a Bio-inspired Technique for Anomaly Based IDS AIS was proposed by Hosseinpour, et al., [11]. This was a new bio-inspired model used to solve problems in information security. AIS unique features encouraged researchers to employ such techniques in various applications especially in IDS. This procedure presented a survey on current AIS based IDS.

IDS adapted from agent based artificial immune systems were proposed by Oil, et al., [12]. A multi Agent Based IDS (ABIDS) inspired by human immune system theory was proposed. Accordingly computer hosts meeting malicious intrusions could be detected through input signals and temporary output signals like PAMP danger and safe signals.

An artificial immune system based IDS was proposed Shen and Wang [13] where many feature sets were tried and compared. A compromise between complexity and detection accuracy was discussed.

A Multi-Agent-Based Distributed IDS which described the development trend of Distributed IDS Shortcomings and advantages of Structure and Principle of work of multi-agent technology based Distributed IDS was proposed by Huang, et al., [14].

A study of detector generation algorithms based on artificial immune in IDS was proposed by Chen, et al., [15] where binary matching rules, listed characteristics were analyzed. Detector generation algorithm was divided into three processes including gene library, negative selection and clone selection.

A GA and AIS was proposed by Sridevi and Chattemvelli [16] which described a combinational approach to network intrusion detection. It suggested investigation of genetic search methods efficiency for featured selection and Immune system to enable classification of threats and non-threats.

Deng and Gao [17] proposed Research on Immune Based Adaptive IDS Model. In this proposed method an Immune based Adaptive Intrusion Detection System Model (IAIDSM) was described. Analyzing the training data got from the Internet the self-behavior set and non-self-behavior set could be obtained.

Distributed agents model for intrusion detection established on AIS was proposed by Yang, et al., [18] with the experimental results showing that the suggested model had features of real-time processing providing a solution for network surveillance.

For MANET a Context Adaptive Intrusion Detection System was proposed by Cheng and Tseng [19] where Context Adaptive Intrusion Detection System (CAIDS) adapted to current node to accommodate and inspect new packets. Performance was evaluated with a reward function that located an effective way to perform intrusion detection. It also delivered security benefits while meeting the energy budget. Numerical results reveal that CAIDS was a good tradeoff between life performance and security. The study demonstrated empirically that CAIDS model intelligently monitored and recognized security breaches attempted while adhering to resource budget over anticipated network life.

An Efficient Formal Framework for IDS was proposed by Rouached and Sallay [20] which improved network intrusion detection process efficiency by including an Event Calculus based specification to detect the network's registered and expected behavior. But, detecting earlier unseen attacks was important to minimize losses due to successful intrusion. It was also important to detect attacks early to minimize impact.

A Performance Evaluation Study of IDS was proposed by Alhomoud, et al., [21] which tested and analyzed performance of IDS system Snort and new IDS system Suricata both of which were implemented on three platforms to simulate real environment. Finally results and analysis were compared and recommendations as to what the ideal environment for Snort and Suricata would be.

An adaptive genetic-based signature learning system for intrusion detection was proposed by Shafi and Abbass [22] which presented a biologically inspired computational approach to learn signatures for network intrusion detection using supervised learning classifier system. The classifier was an online and incremental parallel production rule based system. Performance of developed systems was evaluated with an available intrusion detection dataset and results presented proved the suggested system's effectiveness.

For intrusion detection in computer networks a parallel genetic local search algorithm was proposed by Abadeh, et al., [23] that described a parallel genetic local search algorithm to generate fuzzy rules to detect computer networks intrusive behavior. The system used Michigan's approach where individuals represented a fuzzy rule with the form "if condition and then prediction". In the algorithm the global population was divided into subpopulations each assigned a distinct processor. Each subpopulation had same class fuzzy rules which evolved independently in a parallel manner. Experiments showed that the algorithm produced fuzzy rules to construct reliable IDS.

Intrusion detection in MANET using classification algorithms was proposed by Otrok, et al., [24]. The cost and model selection Intrusion detection was a second line of defense in MANETs. This investigation examined the proper use of classification methods for intrusion detection in MANETs. For this, it evaluated five supervised intrusion detection classification algorithms on many metrics. This method measured performance on a dataset with results indicating that weighted cost matrices were effective with statistical classifiers and that sequential cross validation had small but significant effect for certain classifiers.

A game-theoretic intrusion detection model for MANETs was proposed by Pastrana, et al., [25]. This study addresses issues of increasing effectiveness of IDS for an ad hoc networks node cluster. To lower IDS performance overhead, a leader node is elected to handle intrusion detection for the entire cluster. Experiments reveal that presence of selfish nodes significantly lowers effectiveness of ID as fewer packets are inspected. This article compared effectiveness of six different classifiers in MANETs to detect malicious activities. Results show that Genetic Programming and Support Vector Machines (SVM) may detect malicious activities in MANETs.

An Ensemble method for anomaly detection, and distributed intrusion detection in MANETs was proposed by Cabrera, et al., [26]; this study investigates issues of distributed intrusion detection in MANET using ensemble methods. The new method describes clustering algorithms to update cluster centers and machine learning algorithms to compute local anomaly indexes. The complete algorithms suite was implemented and tested for two MANET routing protocol types and two attacks types against routing infrastructure. Overall results confirmed theoretical developments related to benefits of averaging with detection accuracy improving when it moved up in the node cluster manager hierarchy.

Clark Evolutionary computation techniques for intrusion detection in MANETs was proposed by Sen and Clark [27] that explored use of evolutionary computation techniques specifically, genetic programming and grammatical evolution to bring up ID programs for challenging environments. Aware of power efficiency's importance, it analyzed power consumption, employing a multi objective evolutionary algorithm to locate optimal tradeoffs between intrusion detection ability and power consumed.

Implementation of IDS in response to securing MANETs was proposed by Mohamed and Abdullah [28] that presented a new approach to secure MANETs based on Artificial Immune System (AIS) paradigm. IDS responded to many simulated immune system features. A large amount of immune properties were mapped to ad hoc networks seeking a comprehensive security system. Negative selection, clonal selection, danger theory, immune network concept were main paradigms that were considered. Anomaly detection, first and second response learning capability, decentralized and reliable detection system were features expected to contribute much too ad hoc network security.

Trust Evaluation and Reputation Exchange (TEREC) for Cooperative Intrusion Detection in MANETs was proposed by Ebinger and Bissmeyer [29], which split reputation information into two values; trust and confidence allowing each node to successively determine other nodes reliability without relying on a static, pre-established trust infrastructure needing much overhead. TEREC was evaluated through simulation, and performance measured through the presence of a huge number of malicious nodes. Evaluation results proved that a benign nodes majority prevailed over malicious attacking nodes as they accurately classified network nodes based on reputation estimation.

## Methodology

In this work, NSL KDD dataset was investigated using the proposed Genetic Based Feature Extraction Technique followed by Artificial Immune System Based Classifier. Section 3.1, 3.2 and 3.3 discuss the techniques used. Figure 3 shows the flow of the proposed IDS system

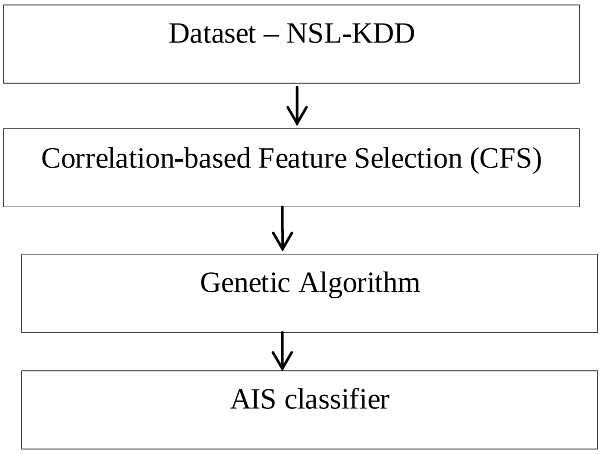

**Figure 3:** Flow Chart of Proposed Method

**NSL KDD Dataset**:
KDD data is from 1998 DARPA IDS. Sponsored by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory (AFRL), MIT Lincoln Labs collected and distributed datasets for evaluation of computer network IDS.

NSL-KDD Database: Tavallaee et al., [30] proposed the NSL-KDD dataset. NSL-KDD dataset is a lower version of original KDD 99 dataset having features similar to KDD 99 which in turn has 41 features and one class attribute. The latter has 21 classes under four attack types: User to Root (U2R) attacks, Probe attacks, Denial of Service (DoS) attacks and Remote to Local (R2L) attacks. This dataset has a binary class attribute. It has also a reasonable training/test instances numbers makes it practical to handle experiments on [31].

The following are the differences between NSL-KDD and original KDD 99 dataset [31-33]:

- It has no redundant records in train set, so classifiers are not biased towards frequent records.
- The proposed sets have no duplicate records; hence learner's performance is not biased by methods with better detection rates on frequent records.

The selected records number from difficulty level group is inversely proportional to records percentage in original KDD 99 data set.

The records numbers in train and test sets are reasonable making it affordable to run experiments on complete set without randomly selecting a small portion. Thus, evaluation results of different research works are consistent and comparable [34].

To evaluate the new method, UDP data available in KDD 99 dataset is used. The UDP stream has 3 different intrusions types namely teardrop, satan and nmap. Teardrop is DoS attack, where legitimate users are denied access to a machine or the attacker uses computing or memory resources.

Two specially fragmented IP datagrams identify the teardrop. Nmap and satan are network probes, mapping machines and services on a network and looking for vulnerable attack points. Nmap scans are characterized by network packets sent to many ports in a time window. A satan scan is recognized by consistent network traffic pattern created by the program. This work aims to evaluate the efficiency of Genetic Algorithm.

In this method, genetic search methods are used to select features and Artificial Immune System (AIS) is used for system classification.

**Correlation-based Feature Selection (CFS)**
CFS algorithm relies on a heuristic to evaluate merit of a features subset. This heuristic considers individual features usefulness to predict class label with level of inter correlation among them. The heuristic is based on the hypotheses which can be stated thus:

"Good feature subsets have features highly correlated to (predictive of) the class, yet uncorrelated to (not predictive of) each other".

Following same directions, Genari states "Features are relevant if values vary systematically with category membership [35]." Thus, a feature is useful when correlated with or predictive of a class; or else it is irrelevant. Empirical evidence from feature selection literature reveals that, with irrelevant features, redundant information should also be eliminated. A feature is redundant if one or more other features are correlated to it. These definitions for relevance and redundancy result in the idea that best features for a specific classification are those which are correlated with one of classes having an insignificant correlation with rest of features in a set.

If correlation between each component in a test and outside variable is known, and inter-correlation between every components pair is given, then correlation between a composite of summed components and outside variable are predicted from:

$$r_{zc} = \frac{k\overline{r_{zi}}}{\sqrt{k + k - (k-1)\overline{r_{ii}}}}$$

Where $r_{zc}$= correlation between the summed components and the outside variable.
$k$ = number of components (features).
$\overline{r_{zi}}$ = average of the correlations between the outside variableand the components.
$\overline{r_{ii}}$ = average inter-correlation between components.

**Genetic Algorithm**

The GA design has components like population initialization, genetic representation [36], selection scheme, fitness function, crossover and mutation. A routing path has nodes sequence in network. The GA is applied to paths obtained from route discovery. A routing path is encoded through a positive integer's string indicating IDs of network nodes. The string length should not be more than number of network nodes.

*Initial population*

Each chromosome represents a potential solution in a GA and this can have more than one solution to begin with. Paths from route discovery phase are considered initial chromosomes.

*Fitness function*

An obtained solution quality is evaluated accurately through the help of fitness function. The aim of using GA is to locate shortest path, lowest throughput between source and destination and large buffer size that a path has. Obtaining shortest path and lowest delay time is of primary concern as then selection is based on buffer size. Fitness of every chromosome is calculated as,

$$f(ch_i) = \left[\sum_{I \in p(s,r)} c_i + c_d\right]$$

The $ch_i$ represents chromosome fitness value and $c_d$ delay time of each chromosome where $c_i$ represents path cost. The above fitness function is maximized involving only shortest path and delay constraints, as buffer size for all paths is checked by the evolutionary process.

*Crossover*

This operator randomly chooses a locus exchanging subsequences before and after the locus between 2 chromosomes to create 2 offspring. For example, strings 10000100 and 11111111 could be crossed over after third locus in each to result in two offspring 10011111 and 11100100. The crossover operator mimics biological recombination between two single−chromosome (haploid) organisms.

*Mutation*

Global searches are mutations. A mutation probability is predetermined before starting the algorithm and applied to every individual bit of every offspring chromosome to determine whether it should be inverted [37]. The pseudo code for the GA is as follows.

```
1. Create a population of random candidate solutions named pop.
2. Until the algorithm termination conditions are met, do the following (each
   iteration is called a generation):
   a) Create an empty population named new-pop.
   b) While new-pop is not full, do the following:
      i.  Select two individuals at random from pop so that individuals which
          are more fit are more likely to be selected.
      ii. Cross-over the two individuals to produce two new individuals.
   c) Let each individual in new-pop have a random chance to mutate.
   d) Replace pop with new-pop.
3. Select the individual from pop with the highest fitness as the solution to the
   problem.
```

**Artificial Immune System (AIS)**

The Biological Immune System (BIS) is a multilayer protection system with each layer providing different types of defense mechanisms to detect, recognize and respond. The functionality of the BIS is investigated and that is how a body restricts itself from invasion of external microorganisms. The AIS is developed by following BIS principle. AIS algorithm’s four forms reported in the literature are negative selection, immune network model, danger theory [38], and clonal selection.

Artificial Immune Systems (AIS) are algorithms and systems inspired by the human immune system which is robust, de-centralized, error tolerant and adaptive. These properties are desirable for developing new computer systems. Contrary to other bio-inspired techniques like GA and neural networks, AIS encompasses algorithms existing because various algorithms implement different properties on various cells. All AIS algorithms mimic behavior and properties of immunological cells, specially T-cells, B-cells, and Dendritic Cells (DCs). The resultant algorithms exhibit varied complexity levels and perform many tasks.

The major portion of AIS work till date is development of three algorithms derived from simplified models; negative selection, clonal selection and immune networks. First-generation AIS algorithms revealed much limitation when used on realistic applications. Hence, a second AIS generation is raising using models from cutting-edge immunology as their base, and is not just mechanisms from basic models [39].

### Ais Basic Concepts

*Initialization / Encoding*

Four decisions have to be made to implement basic AIS: Similarity Measure, Encoding, Mutation and Selection. Once an encoding is fixed and suitable similarity measure chosen, the algorithm then performs selection and mutation, based on similarity measure, till stopping criteria are met.

*Similarity or Affinity Measure*

Similarity measure or matching rule is an important design choice in AIS algorithm development and closely linked to the encoding scheme. Two simple matching algorithms are explained using binary encoding: Consider strings (00000) and (00011). In a bit-by-bit comparison, first three bits are identical and so this pair is given a matching score of 3. In other words, the opposite of Hamming Distance is computed.

*Negative, Clonal or Neighborhood Selection*

This step’s meaning differs depending on exact problem the AIS are applied to. For film recommender, choosing a correct neighborhood means selecting good correlation scores and so ‘positive’ selection is performed.

*Somatic Hyper mutation*

A commonly used mutation in AIS is similar to that in GA, e.g. bits are flipped for binary strings, one value is changed at random for real value strings, or for others the elements order is swapped. Additionally, the mechanism is often enhanced by ‘somatic’ idea, that the closer the match, the more (or less) disruptive is the mutation [40].

### Pseudo code of an Artificial Immune Algorithm

```
1. g = 1 //First Generation
2. Pg.CreateNewAntibodies(PopSize) /
            /Create new PopSize antibodies and evaluate them
3. For each antibody Ai in Pg //Repeat for all antibodies in the population
4. Repeat till Step 8 for NoC times /
            /NoC is a parameter representing the number of clones
5. M = Ai.Clone( ) //Clone Ai
6. M.Mutate( ) //Mutate the clone
7. M.Evaluate ( ) //Evaluate the mutated clone
8. TP.Add( M ) //Add the mutated clone to a temporary population TP
9. BM = TP.GetBest( ) //Get the mutated clone with the best fitness in TP
10. If BM.fitness better than Ai.fitness then Ai
            = B //If best mutated clone better than its parent:   BM replaces Ai
11. TP.Clear() //Delete all antibodies from TP
```

```
12. Next Antibody //Return to step 3 to get another antibody to mutate and clone
13. Pg.Affinity() //Calculate the affinity between each 2 antibodies in Pg
14. $Pg + 1$.Add( Pg.Select()) /
        /Based on the affinity, select antibodies from Pg for $Pg + 1$
15. $Pg + 1$.CreateNewAntibodies(PopSize − $Pg + 1$.Size) //Add new antibodies to Pg
        $+ 1$ to replace  eliminated ones
16. $g = g + 1$ //Next generation
17. If $g >$ IterationsNumber Then Stop Else Go to Step 3 /
        /Stopping criteria based on the number of iterations
```

## Results and Discussion

The study focuses on IDS for efficient classification of normal and abnormal behaviour of the network traffic data and generation of automatic intrusion rules. In the proposed method, Genetic search methods is used for feature selection and Artificial Immune System (AIS) is used for classification of the system. The proposed feature selection method is compared for parameters such as classification accuracy, Root Mean Square Error (RMSE), precision and recall.

The parameters used in the proposed method are

**GA parameters:**

Initial population - 20
Crossover probability - 0.9
Mutation probability - 0.01
Selection using roulette wheel encoding

**AIS parameters**

AIS affinity threshold - 0.2
Hypermutation rate - 2
Initial pool size - 1
stimulation value - 0.9

Figure 4-7 shows the results obtained from the proposed method.

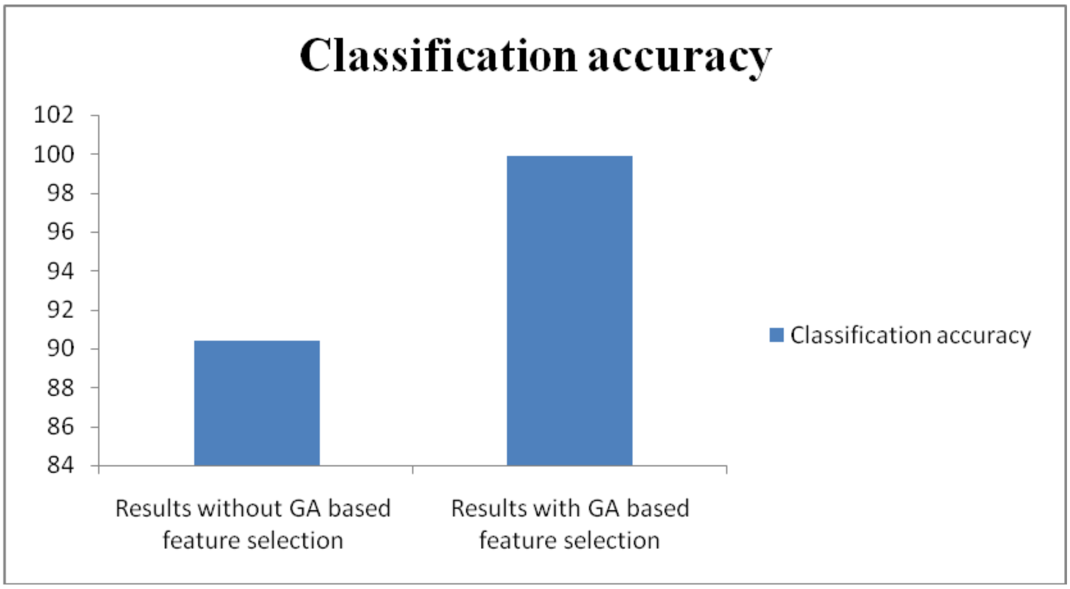

**Figure 4:** Classification accuracy

From figure 4, it is observed that the proposed GA with AIS method provides 99.8793 % classification accuracy than normal GA method.

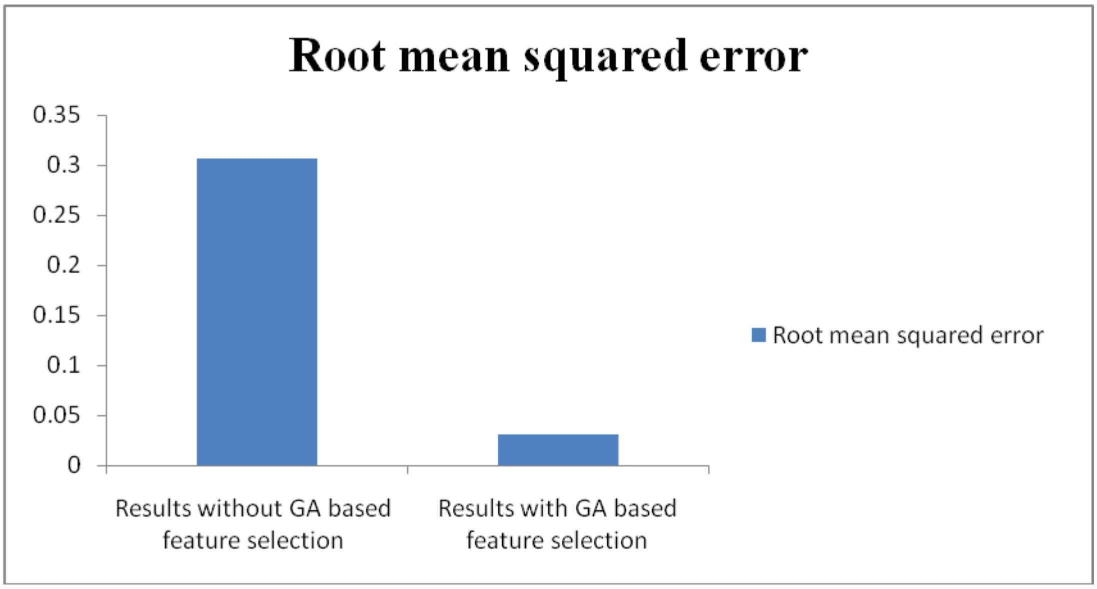

**Figure 5:** Root mean square error

The RMSE value of the proposed GA using AIS is 0.0313 which is less compared to normal GA.

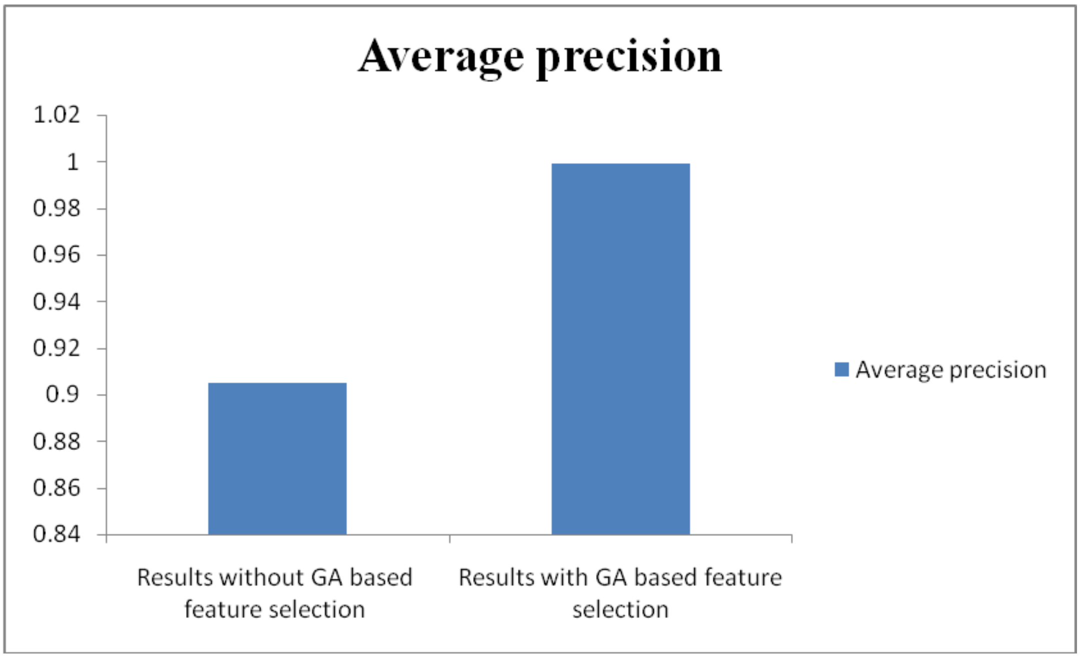

**Figure 6:** Average Precision

The proposed GA with AIS has a high average precision of 0.999% where normal GA has 0.905%.

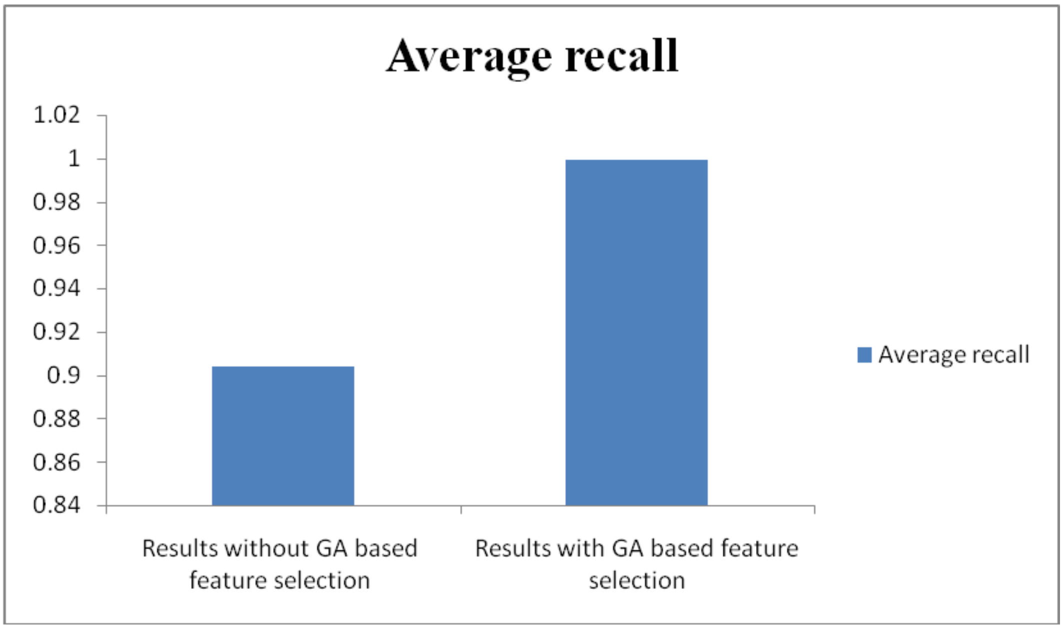

**Figure 7:** Average recall

The proposed GA with AIS has a high average recall of 0.999% where normal GA has 0.904%.

## Conclusion

The study focuses on IDS for efficient classification of normal and abnormal behaviour of the network traffic data and generation of automatic intrusion rules. In the proposed method genetic search methods is used for feature selection, and Artificial Immune System (AIS) is used for classification of the system. The parameters used in the proposed method are GA parameters**:** Initial population – 20, Crossover probability - 0.9, Mutation probability - 0.01, selection using roulette wheel encoding. AIS parameters**:** AIS affinity threshold -0.2, Hyper-mutation rate - 2, Initial pool size - 1, stimulation value - 0.9 by using this parameter the results obtained are as follows. Without GA based feature selection classification accuracy 90.3829%, Root mean squared error 0.3058%, Average precision 0.905%, Average recall 0.904 %. With GA based feature selection Classification accuracy 99.8793%, Root mean squared error 0.0313%, Average precision - 0.999%, Average recall - 0.999%.

## Reference

[1] A. Micksch (2000). Information systems risk analysis, assessment and management. *Global Information Assurance Certification Paper, SANS Institute InfoSec Reading Room*.

[2] A. Sundaram (1996). An introduction to intrusion detection. *Crossroads*, *2*(4), 3-7.

[3] V. Chandola, Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv*, *41*(3), 1-58.

[4] L. Wang. Artificial Neural Network for Anomaly Intrusion Detection.

[5] H. Debar, (2000). An introduction to intrusion-detection systems. Proceedings of Connect.

[6] J.M. Estevez-Tapiador, J. M., Garcia-Teodoro, P., & Diaz-Verdejo, J. E. (2004). Anomaly detection methods in wired networks: a survey and taxonomy. *Computer Communications*, *27*(16), 1569-1584.

[7] VK. Kshirsagar, Tidke, S. M., & Vishnu, S. (2012). Intrusion Detection System using Genetic Algorithm and Data Mining: An Overview. *International Journal of Computer Science and Informatics ISSN (PRINT)*, 2231-5292.

[8] D. Dal, Abraham, S., Abraham, A., Sanyal, S., &Sanglikar, M. (2008, June). Evolution induced secondary immunity: An artificial immune system based intrusion detection system. In *Computer Information Systems and Industrial Management Applications, 2008. CISIM'08. 7th* (pp. 65-70). IEEE.

[9] F.N.M. Sabri, Norwawi, N. M., &Seman, K. (2011, December). Hybrid of rough set theory and artificial immune recognition system as a solution to decrease false alarm rate in intrusion detection system. In *Information Assurance and Security (IAS), 2011 7th International Conference on* (pp. 134-138). IEEE.

[10] F. Hosseinpour, Abu Bakar, K., HatamiHardoroudi, A., &FarhangDareshur, A. (2010, November). Design of a new distributed model for Intrusion Detection System based on Artificial Immune System. In*Advanced Information Management and Service (IMS), 2010 6th International Conference on* (pp. 378-383). IEEE.

[11] F. Hosseinpour, Bakar, K. A., Hardoroudi, A. H., &Kazazi, N. (2010, November). Survey on artificial immune system as a bio-inspired technique for anomaly based intrusion detection systems. In *Intelligent Networking and Collaborative Systems (INCOS), 2010 2nd International Conference on* (pp. 323-324). IEEE.

[12] C.M. Oil, Wang, Y. T., &Ou, C. R. (2011, June). Intrusion detection systems adapted from agent-based artificial immune systems. In *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on* (pp. 115-122). IEEE.

[13] J. Shen, & Wang, J. (2011, November). Network intrusion detection by artificial immune system. In *IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society* (pp. 4716-4720). IEEE.

[14] W. Huang, An, Y., & Du, W. (2010, August). A Multi-Agent-based Distributed Intrusion Detection System. In *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on* (Vol. 3, pp. V3-141). IEEE.

[15] J. Chen, Yang, D., &Naofumi, M. (2007). A Study of Detector Generation Algorithms Based on Artificial Immune in Intrusion Detection System.*WSEAS TRANSACTIONS on BIOLOGY and BIOMEDICINE*, *4*(3), 29-35.

[16] R. Sridevi, & Chattemvelli, R. (2012, March). Genetic algorithm and Artificial immune systems: A combinational approach for network intrusion detection. In *Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on* (pp. 494-498). IEEE.

[17] L. Deng, & Gao, D. Y. (2009, April). Research on immune based adaptive intrusion detection system model. In *Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC'09. International Conference on* (Vol. 2, pp. 488-491). IEEE.

[18] J. Yang, Liu, X., Li, T., Liang, G., & Liu, S. (2009). Distributed agents model for intrusion detection based on AIS. *Knowledge-based systems*, *22*(2), 115-119.

[19] B.C. Cheng, & Tseng, R. Y. (2011). A context adaptive intrusion detection system for MANET. *Computer Communications*, *34*(3), 310-318.

[20] M. Rouached, & Sallay, H. (2012). An Efficient Formal Framework for Intrusion Detection Systems. *Procedia Computer Science*, *10*, 968-975.

[21] A. Alhomoud, Munir, R., Disso, J. P., Awan, I., & Al-Dhelaan, A. (2011). Performance Evaluation Study of Intrusion Detection Systems. *Procedia Computer Science*, *5*, 173-180.

[22] K. Shafi, & Abbass, H. A. (2009). An adaptive genetic-based signature learning system for intrusion detection. *Expert Systems with Applications*,*36*(10), 12036-12043.

[23] M. SanieeAbadeh, Habibi, J., Barzegar, Z., &Sergi, M. (2007). A parallel genetic local search algorithm for intrusion detection in computer networks.*Engineering Applications of Artificial Intelligence*, *20*(8), 1058-1069.

[24] H. Otrok, Mohammed, N., Wang, L., Debbabi, M., & Bhattacharya, P. (2008). A game-theoretic intrusion detection model for mobile ad hoc networks.*Computer communications*, *31*(4), 708-721.

[25] S. Pastrana, Mitrokotsa, A., Orfila, A., & Peris-Lopez, P. (2012). Evaluation of classification algorithms for intrusion detection in MANETs. *Knowledge-Based Systems*.

[26] J.B. Cabrera, Gutiérrez, C., & Mehra, R. K. (2008). Ensemble methods for anomaly detection and distributed intrusion detection in mobile ad-hoc networks. *Information Fusion*, *9*(1), 96-119.

[27] S. Sen, & Clark, J. A. (2011). Evolutionary computation techniques for intrusion detection in mobile ad hoc networks. *Computer Networks*, *55*(15), 3441-3457.

[28] Y.A. Mohamed, & Abdullah, A. B. (2010, June). Implementation of IDS with response for securing MANETs. In *Information Technology (ITSim), 2010 International Symposium in* (Vol. 2, pp. 660-665). IEEE.

[29] P. Ebinger, & Bissmeyer, N. (2009, May). Terec: Trust evaluation and reputation exchange for cooperative intrusion detection in manets. In*Communication Networks and Services Research Conference, 2009. CNSR'09. Seventh Annual* (pp. 378-385). IEEE

[30] M. Tavallaee, Bagheri, E. Wei Lu and Ghorbani, A. (2009). A detailed analysis of the KDD CUP 99 data set. Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009), 1-6

[31] A. Shaheen, (2010). A comparative analysis of intelligent techniques for detecting anomalous internet traffic. MSc. Thesis, King Fahd University.

[32] H.F. Eid, Darwish, A. Hassanien, A.E. and Abraham, A. (2010). Principle components analysis and support vector machine based intrusion detection system. In the Proceedings of 10th International Conference on Intelligent Systems Design and Applications (ISDA 2010), Cairo, Egypt

[33] R. Datti, and Verma, B. (2010). Feature reduction for intrusion detection using linear discriminant analysis. (IJCSE) International Journal on Computer Science and Engineering, 2(4), 1072-1078

[34] L.M. Ibrahim, Basheer, D. T., &Mahmod, M. S. (2013). A Comparison Study For Intrusion Database (Kdd99, Nsl-Kdd) Based On Self Organization Map (Som) Artificial Neural Network. *Journal of Engineering Science and Technology*, *8*(1), 107-119.

[35] M.A. Hall, & Smith, L. A. (1999, May). Feature Selection for Machine Learning: Comparing a Correlation-Based Filter Approach to the Wrapper. In*FLAIRS Conference* (pp. 235-239).

[36] D. Sureshkumar, K. Manikandan and M.A.SaleemDurai (2011) Secure On-Demand Routing Protocol for MANET using Genetic Algorithm International Journal of Computer Applications (0975 – 8887) Volume 19– No.8.

[37] B. Mishra, & Patnaik, R. K. (2009). *Genetic Algorithm and its Variants: Theory and Applications* (Doctoral dissertation).

[38] S.J. Nanda, (2009). *Artificial immune systems: principle, algorithms and applications* (Doctoral dissertation).

[39] J. Greensmith, Whitbrook, A., &Aickelin, U. (2010). Artificial immune systems. In *Handbook of Metaheuristics* (pp. 421-448). Springer US.

[40] E.K. Burke, & Kendall, G. (Eds.). (2005). *Search methodologies: introductory tutorials in optimization and decision support techniques*. Springer.