Design of Intrusion Detection System using Fuzzy Class-Association Rule Mining based on Genetic Algorithm

Swati Dhopte
Research Scholar
Department of Computer Engineering,
Vishwakarma Institute of Technology, Pune, India

N. Z. Tarapore
Assistant Professor
Department of Computer Engineering,
Vishwakarma Institute of Technology, Pune, India

ABSTRACT

Now security is considered as a major issue in networks, since the network has extended dramatically. Therefore, intrusion detection systems have attracted attention, as it has an ability to detect intrusion accesses effectively. These systems identify attacks and react by generating alerts or by blocking the unwanted data/traffic. The proposed system includes fuzzy logic with a data mining method which is a class-association rule mining method based on genetic algorithm. Due to the use of fuzzy logic, the proposed system can deal with mixed type of attributes and also avoid the sharp boundary problem. Genetic algorithm is used to extract many rules which are required for anomaly detection systems. An association-rule-mining method is used to extract a sufficient number of important rules for the user's purpose rather than to extract all the rules meeting the criteria which are useful for misuse detection. Experimental results with KDD99 Cup database from MIT Lincoln Laboratory show that the proposed method provides competitively high detection rates compared with crisp data mining.

Keywords

Data Mining, Intrusion Detection System (IDS), Genetic Algorithm (GA), Network Security, Fuzzy Logic.

1. INTRODUCTION

In recent years, computer security has become increasingly important and an international priority. This is due to the emergence of electronic commerce, the tremendous use of computers and the rapid growth of computer networks. Computer security is defined as the protection of computing systems against threats to confidentiality, integrity, and availability [3]. Security threats come from different sources such as natural forces (flood), accidents (fire), failure of services (power) and people known as intruders. Two types of intruders are: the external intruders who are unauthorized users of the machines who attack by using various penetration techniques, and internal intruders, refers to those with access permission who wish to perform unauthorized activities [2].

When an intruder attempts to break into an information system or performs an action not legally allowed, this activity is referred to as an intrusion. Intrusion techniques may include password cracking, exploiting software bugs and system misconfiguration, sniffing unsecured traffic, or exploiting the design flaw of specific protocols. An IDS is a system for detecting intrusions and reporting to the proper authority.

There are two generally accepted categories of intrusion detection techniques: misuse detection and anomaly detection [4]. Misuse detection mainly searches for specific patterns or sequences of programs and user behaviours that match well-known intrusion scenarios. Anomaly detection develops models of normal network behaviours, and new intrusions are detected by evaluating significant deviations from the normal behaviour. It is stronger than misuse detection; because it has the ability to detect unseen attacks.

IDSs can also be divided into two groups depending on where they look for intrusive behaviour: Network-based IDS (NIDS) and Host-based IDS [4]. Network-based IDS refers to systems that identify intrusions by monitoring traffic through network devices (e.g. Network Interface Card, NIC). Host-based IDS requires small programs to be installed on individual systems to be monitored.

IDSs can also be divided into two groups depending upon behaviour of IDS: Passive and Active type of IDS. Passive IDS simply detects and alerts the administrator whereas, active IDS will not only detect suspicious or malicious traffic and alert the administrator, but will take predefined proactive actions to respond to the threat.

2. RELATED WORK

Methods for IDS are classified as follows:

1. Supervised Learning-Based Approach
2. Unsupervised Learning-Based Approach
3. Data-Mining-Based Approach

Data mining techniques that have been applied to Intrusion Detection Systems are feature selection, machine learning, and statistical techniques.

Feature Selection, in which important features are selected from a large set of features, is a process commonly used in machine learning [24]. The Intrusion Detection concept was introduced by Anderson J. in 1980 [5]. He applied a statistical method to analyze user's behaviour and to detect an attack. Denning D. [6] describes the use of statistical techniques to detect anomalies, also some of the problems and their solutions in such an approach. The idea of intrusion detection system was known progressively, and his paper was regarded as a significant landmark in this area. In contrast to statistical techniques, machine learning techniques are well suited to learning patterns with no apriori knowledge of what those patterns may be. Clustering and Classification are probably the two most popular machine learning problems [29], [7].

In Classification Techniques, classify the instances of dataset into a particular class i.e. either normal or malicious. The challenge in this method is to minimize the number of false positives and false negatives [7]. Five general categories of techniques have been tried to perform classification for intrusion detection purposes [9]:

a) Inductive Rule Generation: The RIPPER (rule learning program) System is the most popular representative of classification techniques. Lee W. et al. [10] used this system and proposed a framework for intrusion detection using data mining techniques. The attractive property of this process is that multiple rule sets may be generated and used with a meta-classifier. As this method uses a greedy search, the improvement process in the algorithm is expensive.

b) Neural Networks: As neural networks do not require any explicit user model, this technique provides a solution to the problem of modeling the users' behavior in anomaly [9]. McHugh J. [11] has pointed out that advanced research issues on IDSs should involve the use of pattern recognition and learning by example approaches.

c) Support Vector machines (SVM): SVM is a machine learning method and is widely applied to the field of pattern recognition [6]. It is also used for an intrusion detection system. "one class SVM" is based on one set of examples belonging to a particular class and no negative examples rather than using positive and negative examples. Neither of these approaches addresses the reduction of the training time of SVM, which is what prohibits real-time usage of these approaches. With regard to the training time of SVM, random sampling has been used to enhance the training of SVM.

d) Genetic Algorithms: In intrusion detection, the GA is employed to derive a set of classification rules from network audit data, and the support-confidence framework is utilized as a fitness function to judge the quality of each rule. Good properties of GA are it is robust to noise, self learning capabilities, no gradient information is required to find the global optimal or sub-optimal solution [30]. High attack detection rate and low false-positive rate are the advantages of GA techniques [13]. Genetic algorithm uses a string structure for representation of rules. A string representation increases the overhead of rule formation that is the overhead for more number of rules generation [12]. In [3], a rule evolution approach based on detecting novel attacks on networks is presented and three genetic operators, namely reproduction, mutation and crossover are used to evolve new rules. Crosbie M. et al. [16] shows genetic programming (GP) which improves the interpretability of GA by replacing the gene structures with the tree structures, which enables higher representation ability of association rules. Due to the use of the tree data structure for rule formation, reuse of many nodes is not possible. So, GP is not a very efficient method for rule mining.

e) Fuzzy Logic: It is derived from fuzzy set theory dealing with reasoning that is approximate rather than precise. Florez G. et al. [26] applied an improved algorithm of the fuzzy data mining approach to the IDS. The fuzzy data mining technique is used to extract the patterns that represent normal behavior for intrusion detection. Luo J. [15] also attempted classification of the data using Fuzzy logic rules.

Desheng F. et al. [17] and Barbara D. et al. [18], is mostly focused on the Association-Rule-Mining algorithm which is widely used for rule mining in Data-Mining. Audit data analysis and mining (ADAM) has the potential to accommodate a large amount of network audit data that keeps growing in size. As encoding rules is time-consuming and highly depends on the knowledge of known intrusions, Zhang J. et al. [19] use a new systematic framework that applies a data mining algorithm called random forests which is a hybrid-network-based IDS. Main purpose of this algorithm is to find the minority attacks which sometimes create more damage than majority attacks. Association rule mining method for intrusion detection, [17], [18], [19] suffers from the sharp boundary problem. Therefore, to overcome this problem fuzzy set theory is used. The system uses a data miner that integrates Apriori and Kuok's algorithms to produce fuzzy logic rules that capture features of interest in network traffic [1], [15]. The limitation of this approach is that, it is not able to work for a mixed database, also more overhead for extraction of rules. In order to discover interesting rules from a dense database, genetic algorithm (GA) and genetic programming (GP) have been applied to association-rule mining.

Hybrid Approach:

This approach is a hybrid approach which was genetic algorithm, fuzzy logic and class-association rule mining algorithm. Due to a hybrid approach, this proposed system works for both misuse and anomaly intrusion detection system.

3. BACKGROUND STUDY

3.1 Datasets

KDD99 CUP is the dataset prepared for the Third International Knowledge Discovery and Data Mining (KDD) Tools Competition, which was held in conjunction with KDD99 the Fifth International Conference on KDD [24]. The competition task was to build a network intrusion detector, a predictive model capable of distinguishing between "bad" connections, called intrusions or attacks, and "good" normal connections. This database contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment. There are approximately 4,940,000 kinds of data in the training dataset, 10% of which is provided, there are 3,110,291 kinds of data in test dataset, and there are totally 41 types of network connection characteristics (characterized by continuous data and discrete data) in each kind of network connection record [2], [24], [25].

3.2 Data Mining

Data mining generally refers to the process of extracting or mining knowledge from a large amount of data. This process first understands the existing data and then predicts the new data. It is the core of Knowledge Discovery and Data mining (KDD). Kind of Patterns found in Data Mining Task are specified by Data Mining Functionalities. In general, data mining tasks are categorized into two categories: predictive and descriptive. The general properties of the data in the database are characterized by Descriptive mining. Inference on the current data in order to make predictions is performed by Predictive mining [29]. Well-known data mining techniques are:

- Classification
- Clustering
- Association-Rule mining

The recent rapid development in data mining contributes to developing a wide variety of algorithms suitable for network-intrusion-detection problems. As one of the most popular data mining methods, association-rule mining is used to discover association rules or correlations among a set of attributes in a dataset. An association rules are used to represent the relationship between datasets and expressed by $A \Rightarrow B$, where A and B contain a set of attributes. This means that if a tuple satisfies A, it is also likely to satisfy B [29]. A typical example of association rule mining is market basket analysis [29].

Let $J = \{i_1, i_2 \ldots i_m\}$ be a set of items. Let Z be a set of database transactions where each transaction T is a set of items such that $T \subseteq J$. Let A be a set of items. A transaction T is said to contain A if and only if $A \subseteq T$. An association rule is in the form $A \Rightarrow B$, where $A \subset J$, $B \subset J$, and $A \cap B = \phi$. The rule $A \Rightarrow B$ holds in the transaction set Z with support s and confidence c in the transaction set Z i.e.,

$$\text{support}(A \Rightarrow B) = P(A \cup B)$$

$$\text{confidence}(A \Rightarrow B) = P\left(B \mid A\right)$$

Rules that satisfy both a minimum support threshold (min_sup) and a minimum confidence threshold (min_conf) are called strong [22], [29]. The overall performance of mining association rules is determined by the first step (explained in [29]). **Apriori** is an algorithm for mining frequent itemsets for Boolean association rules [29]. The name of the algorithm is based on the fact that the algorithm uses prior knowledge of frequent itemset properties. Apriori algorithm employs an iterative approach known as a level-wise search, where k-itemsets are used to explore (k + l)-itemsets. Details about Apriori algorithm is given in [28] and [29].

3.3 Genetic Algorithm

Genetic algorithms (GA) are search algorithms based on the principles of natural selection and genetics, introduced by John Holland in the 1970s and inspired by the biological evolution of living beings. Genetic algorithms abstract the problem space as a population of individuals, and try to explore the fittest individual by producing generations iteratively. Individuals are represented by a string of symbols. Each individual is called a chromosome, and is composed of a predetermined number of genes [23]. The generation of new offsprings includes the operations such as *crossover, mutation* and *selection* operations [9], [21]:

1) Selection: Reproduction (or selection) is an operator that makes more copies of better strings in a new population. Reproduction is usually the first operator applied on a population [9].

2) Crossover: A crossover operator is used to recombine two strings/parents to get better new two strings/children. It is important to note that no new strings are formed in the reproduction phase. In the crossover operator, new strings are created by exchanging information among strings of the mating pool. Types of crossover are explained in [27], [30].

3) Mutation: Mutation adds new information in a random way to the genetic search process [30], [31]. It is an operator that introduces diversity in the population whenever the population tends to become homogeneous due to repeated use of reproduction and crossover operators [30].

3.4 Fuzzy Theory

Crisp sets do not always satisfy the needs of real world applications, because they only allow a membership of 1 or 0, i.e. member or non-member [15]. In the real world, it is not possible at all times to assign an object clearly to a certain group of objects. Rather, it might lie in between two different sets [15]. Therefore, Georg Cantor invented fuzzy sets theory which generalizes member and non-member functions by assigning values that fall in a specified range, typically 0 to 1, to the elements. Fuzzy set theory overcomes the sharp boundary problem by allowing different degrees of memberships [1] and [22].

4. PROPOSED WORK

4.1 Fuzzy Class-Association Rule Mining using Genetic Algorithm

Like most of the existing association-rule mining algorithms, conventional association-rule mining based on GA is able to extract rules with attributes of binary values [1], [20]. However, in real-world applications, databases are more likely to be composed of both binary and continuous values [1]. For extracting the rules with attributes of continuous value, fuzzy set theory is combined with association rule mining algorithm. Fuzzy Class-association-rule mining based on GA method for intrusion detection system overcomes many problems like sharp boundary problem, deals with a mixed database, and increases rule pool size. Therefore, extraction of many rules as compared to other method is possible. Support and fitness factors are calculated for each rule. Fitness function contributes to mining more rules with higher accuracy.

Proposed system objectives are as follows:

- Avoiding the sharp boundary problem by using fuzzy set theory.
- Use of mixed database, increases the detection rate and increases accuracy.
- Increases the size of rule pool by using the genetic operators.
- Flexibly applied to both misuse and anomaly intrusion detection.

4.2 System Architecture and Algorithms for Proposed System

The proposed GA-based intrusion detection using fuzzy data mining approach contains two modules where each works in a different stage. In the training stage, using the GA and fuzzy-association rule mining algorithm, a set of classification rules are generated from KDD dataset. In the intrusion detection stage, the generated rules are used to classify incoming data from a test file. Once the rules are generated, the intrusion detection is simple and efficient. Fig 1 shows the proposed system architecture.

4.2.1 Data Pre-processing

Intrusion detection techniques are misuse intrusion detection and anomaly intrusion detection. For detecting intrusion, rules are required (i.e. rules for normal and attack data). For these, sorting of normal records and attack records from KDD training dataset is required. Inputs of some important features of this sorted dataset are given to the pre-processor. The same pre-processing steps are required for both datasets (i.e. normal and attack dataset). Steps for pre-processing of attributes/features are shown in the following algorithm:

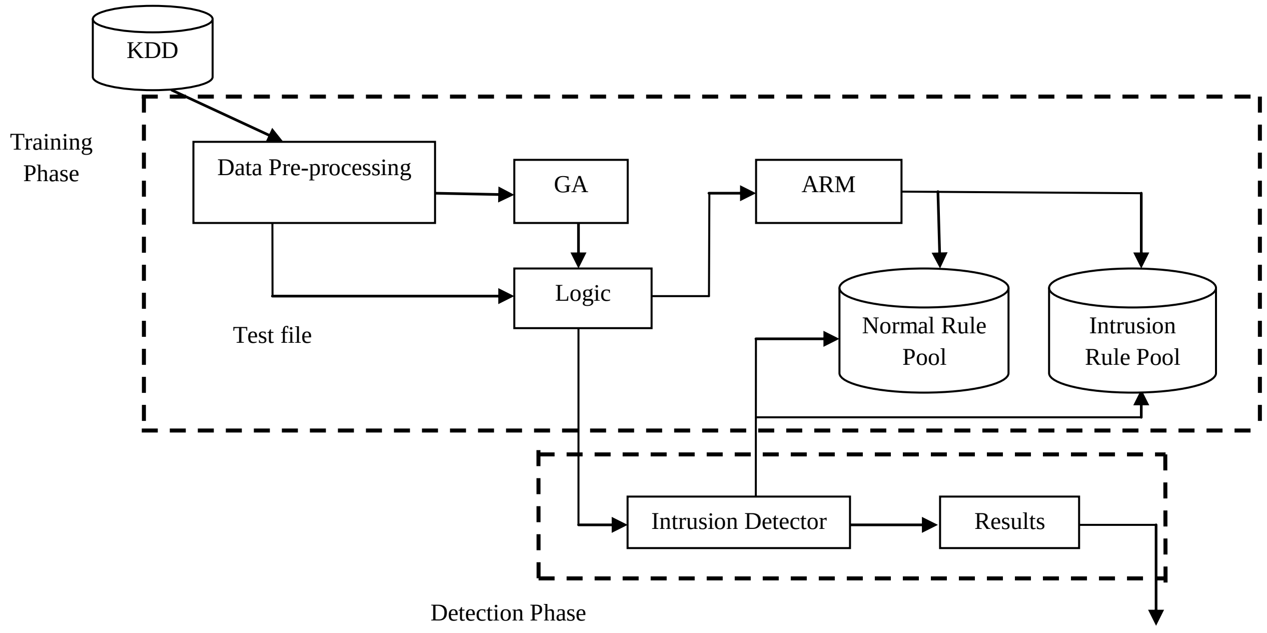


Fig 1: Proposed System Overview

Algorithm: Classify KDD dataset, Feature extraction
Input : KDD dataset
Output : Dataset into two classes i.e. rule pool (Normal and attack)

```
1. Select KDD dataset
2. Classify whole dataset into "normal" and "attack" class
3. Transform attributes to numeric value
4. Find maximum value for each attribute/feature
5. Select important attribute/features
6. Store rules in rule pool
```

In above algorithm, classification method data mining is used for classifying the whole dataset into two classes i.e. "normal" and "attack". Feature selection is necessary because the use all available features are computationally infeasible.

4.2.2 Genetic Algorithm

Data pre-processing algorithm generates rules which are stored in the rule pool i.e. normal rule pool contains normal records and attack rule pool contains records for intrusion. The following algorithm is common for both i.e. normal and attack rule pool and explains about the genetic algorithm and its operators.

Algorithm: Rule pool generation using genetic algorithm.
Input: Pre-processed dataset, number of generations (G), and population size.
Output: Large no. of rules in the Rule pool.

```
1. Initialize the population
2. N is population size, T (minimum fitness value) = 0
3. User input for number of generations (G)
4. Initialize individuals (I) = 1
5. Initialize fitness counter (K) = 1
6. Select two chromosomes (or rules) from population
7. Increment I by 2, K by 1
8. Apply crossover operator to the chromosome
9. Apply mutation operator to the chromosome
10. If rule is present in rule pool then goto step14 for next rule
11. Else
      Calculate the number of connections $Nt_c$ correctly detected by rule $r$
      Calculate the number of connections in the training data $Nt$
      Calculate the number of normal connections $Nn_i$ incorrectly detected by rule $r$
      Calculate the number of normal connections in the training data $Nn$
      Calculate Fitness value of new chromosome
        $\text{fitness}_{r} = \frac{Ntc}{Nt} - \frac{Nni}{Nn}$
12. If fitness is greater or equal to T then,
      Add newly generated chromosome to rule pool
13. Else go to step 14 for next rule
14. Repeat step 10 until K equals to 3
15. Repeat step 5 until I equals to N/2
16. Increment G by 1
17. If number of generations is not reached, go to step 4
18. Display number of rules generated for the input generations
19. Go to next algorithm that is fuzzy rule extraction
```

In the above algorithm, each rule is referred to as a chromosome or individual. In each generation, apply crossover and mutation to increase the number of rules. For Crossover a pair of individuals is determined by first selecting two individuals from the rule pool. A single point crossover is used to reproduce more individuals. In a single point crossover, exchange of genes (attributes value) between two individuals with respect to some point is carried out.

Range of fitness value is [-1, 1], so threshold fitness is 0 in this approach. Once the individuals are selected for making a pair, avoid repeated selection of individuals to make other pairs. The above procedure is then repeated until no individuals for making pairs are remaining. At the end of this algorithm, a large number of rules will be available for further processing. For Anomaly detection, the quantity of rules matters more than quality, whereas for misuse detection quality rules are required. So for both detection systems this algorithm is best suited.

4.2.3 Fuzzy Logic

After applying a genetic algorithm on normal and intrusion rule pool, all possible combinations of rules will be reproduced. On a large dataset, now apply fuzzy logic to avoid the sharp boundary problem. In this module, types of attributes i.e. discrete and continuous are used. For continuous attributes like duration, source bytes, destination bytes, find the maximum values for each attributes and then divide these values into LOW, MEDIUM and HIGH ranges, and find the fuzzy membership value for each attribute. For discrete attributes, numbers of columns are fixed on the basis of types of values for that attribute that is protocol attribute is divided into TCP, UDP and ICMP. The following algorithm shows fuzzy logic implementation for the rule pool.

Algorithm: Fuzzy Rule Extraction.
Input: Normal or Attack rule pool.
Output: Fuzzy Rules in rule pool

```
1.  β = average value of attribute $A_i$; γ = the largest value of attribute $A_i$ in the dataset;
2.  Select features from rule pool
3.  Check for missing entry for all records
4.  Select record from the rule pool
5.  Process all selected attribute
6.  Divide each continuous attribute into LOW, MEDIUM and HIGH
7.  Set fuzzy membership value for each continuous attribute
                $\alpha + \gamma = 2\beta$
8.  Calculate fuzzy membership value for each continuous attribute
9.  Divide each discrete attribute into a number of types
10. Set binary value for each discrete attribute
11. Store all fuzzy rules in fuzzy rule pool
12. Repeat step 5 until all selected columns are covered
13. Repeat step 2 until all records in the rule pool are considered.
```

Each value of continuous attributes in the database is transformed to three linguistic terms (low, middle, and high). A predefined membership function is assigned to each continuous attribute and the linguistic terms can be expressed by the membership function. The parameters *α*, *β,* and *γ* in a fuzzy membership function for attribute A_i is set as follows:

β is average value of attribute A_i in the database
γ is the largest value of attribute A_i in the database

4.2.4 Algorithm for Class-Association-Rule mining (CARM)

After fuzzy implementation, the fuzzy rule pool will be generated and this rule pool is given as an input to association rule mining. Association Rule Mining is a two-step process [22]:

1. Find all frequent itemsets using Apriori algorithm.
2. Generate strong association rules from the frequent itemsets

For rule generation, antecedent part is generated by using apriori algorithm and for consequent; classification method is used in which the whole KDD dataset is distributed into two classes, that is normal and attack class on the basis of labels provided in the dataset. The following algorithm is used for finding the frequent itemsets from the dataset that is Apriori algorithm:

Algorithm: Apriori algorithm for finding frequent itemsets.
Input: Normalize dataset, minimum support (minsupp) = 0.2
Output: Frequent itemsets.

```
1. Initialize I (no. of records) = 1
2. Scan each record of the fuzzy rule pool.
3. Find number of items (N), number of transactions (M)
4. Increment I by 1 and repeat step 2 until last record in the rule pool
5. Initialize k (number of itemset) =1
6. Find frequent itemset $L_k$ from $C_k$ of all candidate itemsets
       Scan D and count each itemset in $C_k$,
       If count is greater than minimum support, then it is frequent
7. Form $C_{k+1}$ from $L_k$; k = k + 1
       Join $L_{k-1}$ itemset with itself to get the new candidate itemsets,
       If found a non-frequent subset then remove that subset.
8. Store frequent itemset in the rule pool
9. Repeat step 6 and step 9 until $C_k$ is empty
```

At the end of above algorithm, the rule pool contains rules which are used for testing of the system. For misuse detection, train the system by giving attack data as an input and form the rules for attack data. For anomaly detection, train the system by giving normal data as an input and form the rules for normal data.

4.2.5 Intrusion Detector Parameter

Detection of attack/intrusion can be measured by following metrics [11]:

- False positive (FP): Corresponds to the number of detected attacks but it is in fact normal.
- False negative (FN): Corresponds to the number of detected normal instances but it is actually as attack, in other words these attacks are the target of intrusion detection systems.
- True positive (TP): Corresponds to the number of detected attacks and it is in fact as attack.
- True negative (TN): Corresponds to the number of detected normal instances and it is actually normal.

The accuracy of an intrusion detection system is measured with respect to detection rate and false alarm rate.

A. Detection rate (DR)

Detection rate refers to the percentage of detected attack among all input test data, and is defined as follows:

$$\text{Detection Rate} = \frac{TP}{TP + TN} * 100$$

B. False Positive Rate (FPR)

False positive rate refers to the percentage of normal data which is wrongly recognized as an attack, and is defined as follows:

$$\text{False Positive Rate} = \frac{FP}{FP + TN} * 100$$

C. False Negative Rate (FNR)

False negative rate refers to the percentage of attack data which is wrongly recognized as normal, and is defined as follows:

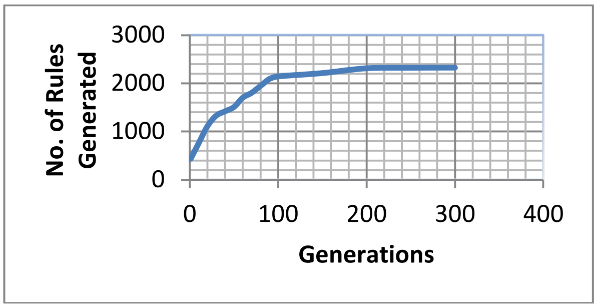
$$\text{False Negative Rate} = \frac{FN}{FN + TP} * 100$$

5. EXPERIMENTAL RESULTS

In this section, the effectiveness and efficiency of the proposed method are studied using KDD 99 Cup dataset.

5.1 Misuse Detection

The proposed method for misuse detection is carried out with KDD 99 Cup database in order to compare results with other machine-learning methods. The training dataset contains 400 attack connections randomly selected from KDD 99 Cup database, where four types of attacks (Dos, Probe, U2R and R2L) are included. A total of 6 attributes are included in each connection; first 6 attributes that is duration, protocol type, service, flag, source bytes and destination bytes respectively. Fig 2 shows that after 250 generations 2323 rules are extracted. Each rule is extracted if it occurs frequently with a statistically significant level in the database. Therefore, each rule is extracted from the whole database by taking into account all the connection data.



Initial Population = 400 individuals

Fig 2 Number of Rules generated in Misuse Detection

The graph above shows after 150 generations rule extraction process decreases and rule extraction stops after 250 generations. Rule generation stops or decreases because the newly generated rule may be already present in the rule pool and according to the proposed algorithm, it avoids redundant data.

Fig 3 shows graph for the number of generations versus time required for execution of algorithm. As the number of generations increase, time required for execution also increases. If initial population increases, then also execution time increases as genetic operators, fuzzy logic and association rule mining requires more number of iterations for processing the initial population. In the graph, for 60, 150 and 250 number of generations, less time is required for execution, because comparatively less rules get extracted. If an extracted rule is already present in the rule pool, then that rule is discarded.

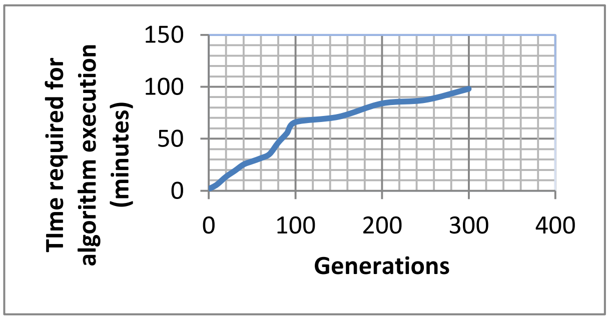


Fig 3 Time required for execution vs no. of generation in Misuse Detection

The testing database contains 500 labeled connections where 400 are labeled intrusion connections (the same types as the training database) and 100 are normal connections. The detection results obtained by the proposed misuse detection classifier are shown in Table 1, where T represents the label of the testing results given by the classifier and C represents the correct label. DR, FPR and FNR are the criteria for evaluation of testing results. The table below shows results after 250 generations of training dataset.

Table 1 Testing Result of Misuse Detection

	Normal (T)	Attack/Intrusion (T)	Total
Normal (C)	98	2	100
Attack/Intrusion (C)	12	388	400
Total	110	390	500

DR= ((98+388)/ 500)*100 = 97.2%

FPR = (2/ 100)*100 = 2%

FNR = (12/ 400)*100 = 3%

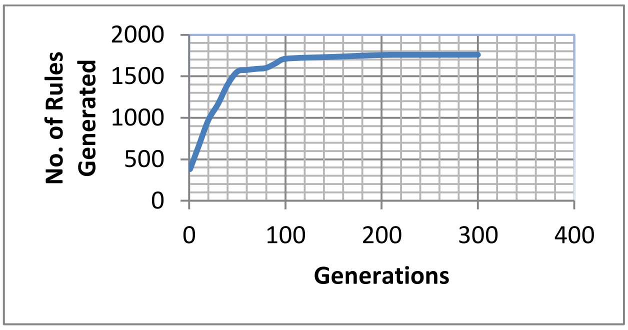
Table 2 shows a comparison between crisp data mining and fuzzy data mining for a misuse detection system. For crisp data mining [1] used 3342 connections randomly from the KDD dataset. For fuzzy data mining only 400 connection are used. The comparison table shows that detection rate for crisp data mining approach is 98.3% by using 3342 initial population. But fuzzy data mining (proposed approach) gives 97.2% result by considering 400 connections as an initial population and 250 numbers of generations. This shows fuzzy data mining method gives more accurate results than other method. For misuse detection, if attack rule pool contains accurate signatures for intrusion, then system will give more accurate result.

Table 2 Comparison of the Detection Rate between Crisp Data Mining and Fuzzy Data Mining in the Misuse Detection

	Crisp Data Mining (%)	Proposed Approach (%)
Detection Rate (DR)	98.3	97.2
False Positive Rate (FPR)	0.67	2
False Negative Rate (FNR)	5.0	3

5.2 Anomaly Detection

The proposed method for anomaly detection is evaluated by using KDD database. The training database is intrusion-free for the purpose of anomaly detection. It contains 350 normal connection records. After 250 generations, 1758 rules related to the normal connections are extracted. Fig 4 shows the number of extracted rules versus number of generations, which indicates that the proposed method can extract rules of the normal connections efficiently through the generations. The graph below shows that after 100 generations, rule extraction process decreases and there is no more new rule extraction after 150 generations. Rule generation stops or decreases because a newly generated rule may be already present in the rule pool and according to the proposed algorithm; it avoids the redundant rules in the rule pool. Gradual increase or decrease of rule extraction depends on the fitness function of the new rule and its existence in the rule pool.



Initial Population = 350 individuals

Fig 4 Number of Rules generated in Anomaly Detection

Fig 5 shows graph for the number of generations versus time required for algorithm execution. In the graph, for 90 and 250 numbers of generations, less time is required for execution, because comparatively less rules are extracted over here than the other generations.

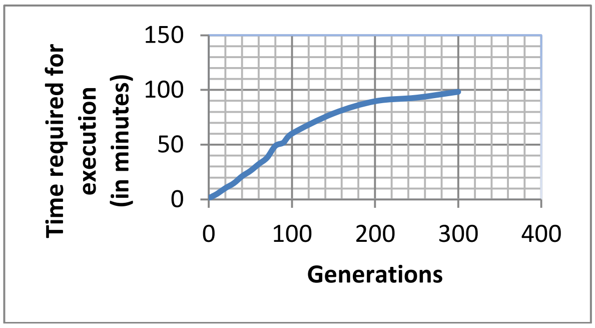


Fig 5 Time required for execution vs no. of generation in Anomaly Detection

The testing database contains 500 connection records including 100 labeled normal records and 400 labeled intrusion records. Since the training database for anomaly detection is intrusion-free (that is normal rule pool), all kinds of intrusions (such as back, ipsweep, land, neptune, pod, port sweep, satan, smurf, and teardrop) are considered unknown/intrusion. The results after 250 generations are shown in Table 3.

Table 3 Testing Result of Anomaly Detection

	Normal (T)	Attack/Intrusion (T)	Total
Normal (C)	97	3	100
Attack/Intrusion (C)	18	382	400
Total	115	385	500

DR= ((97+385)/500)*100 = 95.8%

FPR = (3/100)*100 = 3%

FNR = (18/400)*100 = 4.5%

Table 4 shows a comparison between crisp data mining and fuzzy data mining for a misuse detection system. For crisp data mining [1] used 9137 connections randomly from the KDD dataset. For fuzzy data mining only 350 connections are used. So the result is higher than this for 9137 connections. This shows that the fuzzy data mining method gives more accurate results than other method. For anomaly detection, the system requires more normal rules than accurate; if numbers of rules are many then system will give high detection rate.

Table 4 Comparison of the Detection Rate between Crisp Data Mining and Fuzzy Data Mining in Anomaly Detection

	Crisp Data Mining (%)	Our Approach (%)
Detection Rate (DR)	90.3	95.8
False Positive Rate (FPR)	10.3	3
False Negative Rate (FNR)	9.5	4.5

6. CONCLUSION

Data mining methods are capable of extracting patterns automatically and adaptively from a large amount of data. Various methods related to intrusion detection system are studied and compared. Crisp data mining methods such as ADAM method, Random Forest algorithm are used for intrusion detection but suffer from sharp boundary problem which gives less accurate results. In proposed method, use of fuzzy logic overcomes the sharp boundary problem. Class-Association rules have been used to mine training data to established normal patterns for anomaly detection. An actual intrusion with a small deviation may match the normal patterns and thus not be detected. Therefore, integration of fuzzy logic with class-association rules and GA generates more abstract and flexible patterns for anomaly detection.

In this paper, we have proposed a GA-based fuzzy Class Association Rule Mining with Sub-Attribute Utilization and its application to classification, which can deal with discrete and continuous attributes at the same time. In addition, this method was applied to both misuse detection and anomaly detection. Experiments were performed with practical data provided by KDD99 Cup. The experiment results show that for misuse detection, the proposed method can provides high detection rate and low false positive rate, which are two important criteria for security systems. For anomaly detection, the method provides high detection rate and reasonable false positive rate even without prior knowledge of attack signatures, which is an important advantage over other methods.

7. REFERENCES

[1] Mabu S., Chen C., Shimada K., “An Intrusion-Detection Model Based on Fuzzy Class-Association-Rule Mining Using Genetic Network Programming,” IEEE Transactions Systems, Man, Cybernetics C, Application and Reviews, volume 41, number 1, pp. 130–139, January 2011.

[2] Hoque M., Mukit M. and Bikas M., “An Implementation of Intrusion Detection System using Genetic Algorithm,” International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, March 2012.

[3] Lu W. and Traore I., “Detecting new forms of network intrusion using genetic programming,” Computer Intelligence, volume 20, no. 3, pp. 474–494, 2004.

[4] Kaliyamurthie K., Parameswari D., Suresh R., "Intrusion Detection System using Memtic Algorithm Supporting with Genetic and Decision Tree Algorithms," IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012.

[5] Anderson J., "Computer Security Threat Monitoring and Surveillance," February 26, 1980- revised April 15, 1980.

[6] Denning D., "An intrusion detection model," IEEE Trans. Software Eng., vol. 13, no. 2, pp. 222–232, Feb. 1987.

[7] Ektefa M., Memar S., "Intrusion Detection Using Data Mining Techniques," IEEE Trans., 2010.

[8] Reddy E., Reddy V., Rajulu P., "A Study of Intrusion Detection in Data Mining", Proceedings of the World Congress on Engineering 2011 Vol III WCE 2011, July 6 - 8, 2011, London, U.K.

[9] Shetty M. and Shekokar N., "Data Mining Techniques for Real Time Intrusion Detection Systems," International Journal of Scientific & Engineering Research Volume 3, Issue 4, April 2012.

[10] Lee W. and Stolfo S., "Data Mining Approaches for Intrusion Detection," Computer Science Department Columbia University.

[11] Mchugh J., "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory," ACM Transactions on Information and System Security, Vol. 3, No. 4, November 2000, Pages 262–294.

[12] Naidu N. and Dharaskar R., "An Effective Approach to Network Intrusion Detection System using Genetic Algorithm", International Journal of Computer Applications (0975 - 8887) volume 1 No.2, 2010.

[13] Bankovic Z., Stepanovic D., Bojanic S., "Improving Network Security using Genetic Algorithm Approach," Computer and Electrical Engineering, pp. 438-451, 2007.

[14] Shanmugam B. and Idris N., "Hybrid Intrusion Detection Systems (HIDS) using Fuzzy Logic", Advanced Informatics School (AIS), University Technology Malaysia International Campus, Kuala Lumpur, Malaysia.

[15] Luo J., "Integrating fuzzy logic with data mining methods for intrusion detection," Master's Thesis, Department of Computer Science, Mississippi State University, Starkville, MS, 1999.

[16] Crosbie M. and Spafford G., "Applying genetic programming to intrusion detection," presented at the AAAI Fall Symp. Series, AAAI Press, Menlo Park, CA, Tech. Rep. FS-95-01, 1995.

[17] Desheng F., Zhou S., Guo P., "Research on a Distributed Network Intrusion Detection System Based on Association Rule Mining," The 1st International Conference on Information Science and Engineering (ICISE2009).

[18] Barbara, D., Couto, J., Jajodia, S., & Wu, N., "ADAM: A testbed for exploring the use of data mining in intrusion detection", ACM SIGMOD Record, 30 (4), 15—24, 2001.

[19] Zhang J., Zulkernine M., and Haque A., "Random-forests-based network intrusion detection systems," IEEE Transactions Systems, Man, Cybernetics C, Applications and Reviews, volume 38, no. 5, pp. 649–659, September 2008.

[20] Semaray J., Edmonds J., and Papa M., "Applying data mining of fuzzy association rules to network intrusion detection," presented at the IEEE Workshop Information, United States Military Academy, West Point, NY, 2006.

[21] Abdullah B., Abd-alghafar I., "Performance Evaluation of a Genetic Algorithm Based Approach to Network Intrusion Detection System," 13th International Conference on Aerospace Sciences & Aviation Technology, ASAT- 13, 2009.

[22] Helm B., "Fuzzy Association Rules: An Implementation in R," Master's Thesis, Vienna University of Economics and Business Administration Vienna, 2007.

[23] Gong R., Zulkernine M., Abolmaesumi P., "A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection," Proceedings of the Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Dis tributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks, IEEE, 2005.

[24] Sathya s., Ramani R., Sivaselvi K., "Discriminant Analysis based Feature Selection in KDD Intrusion Dataset," International Journal of Computer Applications (0975 – 8887), Volume 31– No.11, October 2011.

[25] Kddcup 1999data [Online]. Available: kdd.ics.uci.edu/ databases/kddcup99/kddcup99 .html.

[26] Florez G., Bridges S., Vaughn R., "An Improved Algorithm for Fuzzy Data Mining for Intrusion Detection", Annual Meeting of The North American Fuzzy Information Processing Society Proceedings, 2002.

[27] NeuroDimension inc. [Online]. Available: http:// www.nd.com/products/genetic/crossover.html

[28] Agrawal R. and Srikant R., "Fast algorithms for mining association rules," in Proceeding 20th VLDB Conference, Santiago, Chile, pp. 487–499, 1994.

[29] Han J., Kamber M., "Data Mining," Morgan Kaufmann Publishers, 2001.

[30] Goldberg D., "Genetic Algorithm in Search, Optimization and Machine Learning," Reading, MA: Addison-Wesley, 1989.

[31] Koza J., "Genetic Programming, on the Programming of Computers by Means of Natural Selection. Cambridge," MA: MIT Press, 1992.