Discovering vicious nature of network through Automated Discovery of Fuzzy Decision Rules

Garima Ahuja
Department of Computer & Science Engineering
Manav Institute of Technology & Management,Jevra –Hisar , Haryana,India

Rishi
Department of Computer & Science Engineering
Manav Institute of Technology & Management,Jevra –Hisar , Haryana,India

Abstract-With the ballooning of computer networks operation and the immense increase in the number of applications running on crest of it, security in networks is becoming progressively more important. [1] All computer systems are subjected to security vulnerabilities which are both technically troublesome and economically costly to be solved by the manufacturers. That's why Intrusion Detection Systems (IDSs) serves special-purpose to detect anomalies and attacks in the network. Intrusion Detection System (IDS) is a detection technology that preserve data integrity and system availability. Today, there are many commercial Intrusion Detection systems available. But, most of these commercial implementations are ineffective and insufficient. That gives rise to the need of research on this topic.

Keywords- Intrusion Detection System, KDD Cup 99 Dataset, Genetic algorithms, Fuzzy Classification Rules

I. INTRODUCTION

The act of gaining unauthorised access to system is called intrusion. A system that detects intrusion is called Intrusion Detection System. It is also the responsibility of the system to produce reports, alerts and actions. There are various ways to train an Intrusion Detection System. These are host-based and network-based intrusion detection systems. Host-based IDS secure the services of audit logs and system calls as its data source, whereas network-based IDS employs network traffic as its data source. A host-based IDS is made up of an agent on a host which discovers different intrusions by examining audit logs, system calls, file system changes (password files, binaries, etc.), and other related host activities. In network-based IDS, sensors are allocated at strategic position within the network system to apprehend all incoming traffic flows and analyze the contents of the individual packets for intrusive activities such as denial of service attacks, buffer overflow attacks, etc. Each approach has its own potency and weaknesses. Some of the attacks can only be detected by host-based or only by network-based IDS.

KDD Cup 99 Dataset Description

KDD '99 is the most widely used data set used in Intrusion Detection Systems since 1999[2] prepared by Stolfo et al[3] and is built based on the data captured in DARPA'98 IDS evaluation program [4].It comprises approximately 4,900,000 single connection vectors each of which contains 41 features. These higher-level features help in distinguishing normal connections from attacks.

KDD '99 features are classified as:

- Basic features: contains features from a TCP/IP connection.
- Content features: unlike most of the DOS and probing attacks, there appear to be no sequential patterns that are chronic in records of R2L and U2R attacks. This is because the DOS and probing attacks take note of many connections to some host(s) in a very short period of time, but the R2L and U2R attacks are lodged in the data portions of packets, and normally take into account only a single connection.
- Traffic features: contains attributes that are computed relative to a window interval and are further classified as**:**

A. **"same host"** features: analyse only the connections in the past two seconds that have the same destination host as the current connection, and calculate statistics related to protocol behaviour, service, etc.
B. **"same service"** features: analyse only the connections in the past two seconds that have the same service as the current connection.

Types of Attacks

The attacks are classified in the following four categories:

1) Denial of Service Attack (DoS): is an attack in which the attacker attempts to make memory resource too busy or too full to handle legitimate requests, or refuse to give legitimate users access to a machine.

2) User to Root Attack (U2R): is a class of attack in which the attacker starts out with admittance to a normal user account on the system (perhaps gained by a dictionary attack, sniffing passwords, or social engineering) and is able to prey on vulnerability to gain root access to the system.

3) Remote to Local Attack (R2L): occurs when an attacker has the skills to access through remote machine. Eg. Password guessing.

4) Probing Attack: is an attempt to gather information about a network of computers for the apparent purpose of circumventing its security controls.

The main intent behind integrating fuzzy logic (FL) with genetic algorithm (GA) is to cope with real world cognitive uncertainties such as obscurity and ambiguity involved in classification problems. Moreover, instead of using numeric values or ranges in rules, three fuzzy linguistic variables as small, medium and large are used for making the discovered rules more comprehensible.

Genetic algorithms

Genetic algorithms(GAs) are a search heuristic used to find solutions to problems[5]. It does so by mimicking the biological process of natural selection. Operations such as mutation, selection and crossover are used to evolve and improve solutions. Solutions to problems are represented as bit strings. For example, IDS rules can be represented as bit strings. Firstly, a randomly generated population of potential solutions is created. Then mutation, crossover and selection are applied to each generation until an acceptable solution is found or some time limit is exceeded. Mutation is where random bits in an individual or possible solution are randomly changed. Crossover is where two individuals swap sequences of bits to form two new individuals. Selection is where individuals that have better fitness are chosen to parents. Selection combined with a fitness function directs the search towards an effective solution.

Fuzzy Logic

Attacks on systems do not always have a fixed pattern, so fuzzy logic is used to detect patterns that have a behaviour that is between normal and unusual. Fuzzy logic is more like human thinking, based on degrees of truth. Fuzzy rules are represented by if-then statements in the following format: If (condition) then (consequence). The condition part of the rule is composed of one or more features, and the consequence of the rule says if it is an intrusion or not. This process is called fuzzification.

Fuzzy Classification Rules

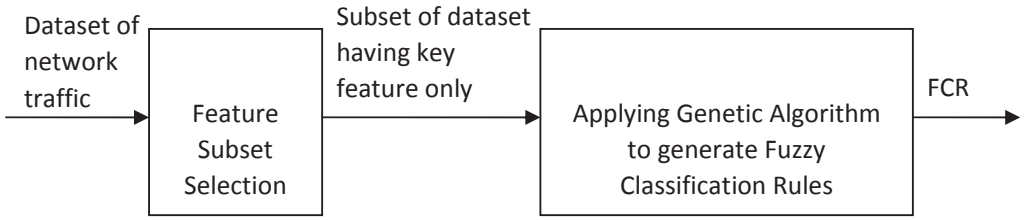
Concept of fuzzy sets can be considered as a generalization of the concept of crisp sets, as in case of fuzzy sets the degree of membership can take on any value in the continuous interval [0...1], whereas in case of crisp sets the degree of memberships can take on only the value 0 or 1. FCR is represented as:

$$IF\ x_1\ is\ A_1\ AND\ x_2\ is\ A_2\ \dots\ AND\ x_n\ is\ A_n$$

$$THEN\ y\ is\ B$$

Where, $X_1, X_2, \ldots, X_n$ denotes the number of attributes in the antecedent part of the rule, y is the target class attribute and $A_1, A_2, \ldots, A_n$ and B are fuzzy membership functions associated with linguistic terms in the antecedent and consequent parts respectively of an FCR[6].

II. PROPOSED WORK



Discovery of FCRs (D)

Input: Dataset D

Output: FCRs

i. Generate at random an initial population of rules representing potential solutions to the classification problem by implementing following steps over the dataset D.
ii. Normalize the population.
iii. Fuzzify the population.
iv. Encode the population.
v. Evaluate each rule on the basis of an appropriate fitness function.
vi. Select the rules to undergo the mechanism of reproduction.
vii. Apply the genetic operators, such as crossover and mutation, to generate new rules.
viii. Reinsert these offspring to create the new current population.
ix. Repeat steps (ii) to (v) until no further improvements occur or a fixed maximum number of generations have been reached.

In proposed system, a Michigan style GA is suggested where each chromosome in the population is a rule. Each rule uses disjunctions between the sub-attributes isolated by conjunctions with other attributes rather than the other works which involved only conjunction between the attributes.

III. EXPERIMENT AND RESULT

KDD training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or an attack. KDD training dataset is very large so here ony 10% KDD training dataset is being used. The feature subset selection is performed on 10% KDD training dataset so that 41 attributes reduces to 8 attributes. Finally the redundant records are eliminated from the dataset using Microsoft Office Excel 2007.The removal of redundant records reduces the number of instances from 494021 to 33517.

Simulation Parameters

Sr. No.	Parameters	Values
1	Population Size (N_{pop})	100
2.	Crossover Probability (P_c)	0.66
3.	Mutation Probability (P_m)	0.1
4.	Maximum Generations(max_gen)	100

The parameters are tuned in the few initial runs of the GAs and the proposed algorithm was terminated when the best Fitness did not change continually throughout 10 generations.

Experimental Results

The desired rule set was discovered by carrying out the simulation process using the simulation parameters. Now this table shows the discovered rule set.

KDD CUP 99 DATASET			
Sr No.	Rules: PRs	Coverage	Fitness
1.	if (((Src_Bytes =Small) || (Src_Bytes =Medium)) && ((Num_Failed_Logins =Small) || (Num_Failed_Logins = Medium)) && (Root_Shell =On) && (Num_Access_Files =Small) && ((Srv_Count =Small) || (Srv_Count =Medium)) && (Serror_Rate =Small)) then Attack.	0.946643	0.667197
2.	if (((Src_Bytes =Small) || (Src_Bytes =Large)) && ((Root_Shell =On) && ((Srv_Count =Small) || (Srv_Count =Medium)) && ((Serror_Rate =Small) || (Serror_Rate =Medium)) then Attack.	0.00144207	0.667197
3.	If (((Duration=Small) || (Duration=Medium)) && ((Src_Bytes =Small) || (Src_Bytes =Large)) && ((Num_Failed_Logins =Small) || (Num_Failed_Logins =Medium) && (Root_Shell =On) && ((Num_Access_Files =Small) || (Num_Access_Files =Large)) && ((Srv_Count=Small) || (Srv_Count =Medium)) && (Serror_Rate=Small) && (Same_Srv_Rate =Large)) then Attack.	0.00546992	0.667197
4.	If (((Duration=Small) || (Duration=Medium)) && (Num_Failed_Logins =Small) && (Root_Shell =On) && (Srv_Count =Small)) then Attack.	0.0362506	0.66718

Predictive Accuracy

The predictive accuracy of the discovered rule sets for different initial population size and maximum generation of the experimental datasets.

Intial Population Size	**Maximum Number of Generations**	**Predictive Accuracy**	**Final number of rules generated**	**Time Taken (in milliseconds)**
50	50	0	1	18979

75	50	95	3	28690
75	75	90	1	41971
75	100	95	4	59997
100	100	98	4	79161
200	200	98	4	452906
500	500	98	5	2450272

IV. CONCLUSION & FUTURE CONSIDERATIONS

In recent years there has been increasing interest in applying Evolutionary Algorithms (EAs) to Knowledge Data Discovery. The work presented in this has demonstrated successful application of GAs for automated discovery of Fuzzy Decision Rules to detect malicious behaviour of a network. The underlying knowledge representation is capable of handling uncertainty and vagueness inherent to decision making support systems.

A genetic approach is proposed for the discovery of decision rules in the form of Fuzzy Classification Rules (FCRs) that can efficiently cope with vague data which do not have crisp boundaries between them. The proposed scheme has adjustable chromosome encoding and appropriate crossover and mutation operators have been described. Keeping in view the basic constraints on FCRs, a well-suited fitness function is formulated so as to make the task of rule mining easier.
The performance of this proposed algorithm is tested across KDD Cup 99 datasets and the results are quite reassuring and have established the effectiveness of the proposal. The scheme provides a mechanism to discover concise and comprehensible classification rules in the form of FCRs. Future work includes creating a standard test data set for the genetic algorithm proposed in the paper and applying it to the test environment.

It generally takes time for techniques to mature and become robust and effective for use in real world problems. There is always some scope for the improvement. The proposed work can also be further explored in the light of the following suggestions:

- *Comparative study of various types of fuzzy rules*

A comparative analysis of the two kinds of fuzzy rule-types Mamdani-type fuzzy rule and Sugeno-type fuzzy rule can be carried out; as in our proposed work we have explored Mamdani-type fuzzy rule only.

- *Discovery of Production Rules with Exceptions*

One of the most important extensions of present work would be development of EAs for the automated discovery of Fuzzy Censored Production Rules (FCPRs) [7] from large datasets.

- *Reducing Time and Space Complexity*

An efficient feature selection method can be used to produce attributes that are more appropriate to produce desired output in a comparative less amount of time or memory.

REFERENCES

[1] C. E. Landwehr, A. R. Bull, J. P. McDermott, and W. S. Choi, "A taxonomy of computer program security flaws," ACM Comput. Surv., vol. 26, no. 3, pp. 211–254, 1994.
[2] KDD Cup 1999. Available on: http://kdd.ics.uci.edu/databases/kddcup 99/kddcup99.html, Ocotber 2007.
[3] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan, "Costbased modeling for fraud and intrusion detection: Results from the jam project," discex, vol. 02, p. 1130, 2000.
[4] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, and M. A. Zissman, "Evaluating intrusion detection systems: The 1998 darpa off-line intrusion detection evaluation," discex, vol. 02, p. 1012, 2000
[5] Goldberg D.E.: "Genetic Algorithms in Search, Optimization and Machine Learning". *New York: Addison-Wesley Publishing Company*, Inc. MA, 1989.
[6] Lotfi A. Zadeh, May 1996, "Fuzzy Logic = Computing with Words", *IEEE Trans. Fuzzy Syst*., vol. 4, no. 2, pp. 103-111.
[7] Saroj and K.K. Bharadwaj, Oct. 2009, "Discovery of Exceptions: A Step towards Perfection", In Proc. 3rd Int. Conf. on Ntwrk. & Syst. Security (NSS), Queensland, Australia, pp. 540-545.