Network Intrusion Detection Using PSO Based on Adaptive Mutation and Genetic Algorithm

Bharat Rathi, Dattatray V. Jadhav

Abstract— The Particle Swarm Optimization is very efficient in intrusion detection in the networks. However, many intrusion detection systems either fail to detect or falsely detect the intrusions. This paper proposes a technique for intrusion detection using Particle Swarm Optimization with Genetic Algorithm based feature selection and using Adaptive Mutation for slow convergence of optimization algorithm. The results thus obtained are approximately 92% that proves the proposed approach to be quite effective in intrusion detection.

Index Terms— Adaptive Mutation, Genetic Algorithm, Intrusion Detection, Particle Swarm Optimization, andKDDCup'99 Data Set.

1 Introduction

Particle Swarm Optimization (PSO), originally introduced by Eberhart and Kennedy [1] in 1995, is a class of random optimization algorithm inspired by swarm intelligence. It has been proved efficient at solving global optimization and engineering problems in [2]. The advantages of PSO over many other optimization algorithms are its implementation simplicity and ability to converge to a reasonably good solution quickly. PSO has been successfully used in many applications, also in the rule extraction for intrusion detection [3]. But a serious problem of IDSs is too many false alarms. Even a small false alarm rate, compared with the daily mass of network traffic, can lead to a large number of alarm information in excess of the number of manual processing level.

Also with the rapid development of the Internet, more and more types of network attacks are founded from the vast network traffic data and it is a time-consuming and labor-intensive work even for an expert. Hence some research work has been carried out on intrusion detection feature selection algorithm to reduce the time required to carry out the computations for intrusion detection. The diversity of different learning algorithms was utilized. Chebrolu, et al. [4] investigated the performance of two feature selection algorithms involving Bayesian Networks (BN) and Classification and Regression Trees (CART) respectively. Both of the approaches considered the diversity of different learning algorithms for intrusion detection. So, the detection performance was promoted accordingly. Optiz [5] presented a genetic algorithm (GA) approach for searching for an appropriate set of feature subsets for ensembles. Using neural networks as the classifier, results showed better than the ensemble approaches of Bagging and Boosting. Tsymbal, et al. [6] presented an algorithm for building ensembles of simple Bayesian classifiers by using different feature subsets generated with the random subspace method.

In order to extract high-quality rules to reduce the false alarm rate of IDS, this paper employs PSO algorithm, Genetic Algorithm for feature selection and adaptive mutation for delaying the convergence of PSO. The experimental results show that the proposed approach is effective and feasible.

- *Bharat Rathi is currently pursuing masters degree program in Electronics & Telecommunication engineering in TSSM's Bhivarabai Sawant College of Engineering & Research, Narhe, Pune under University of Pune, India, PH-09767842908. E-mail:bharat.rathi.88@gmail.com*
- *Dattatray V. Jadhav is PhD in Electronics and is currently holding the post of Principal inTSSM's Bhivarabai Sawant College of Engineering & Research, Narhe, Pune under University of Pune, India. E-mail: dvjadhao@yahoo.com*

2 Standard Particle Swarm Optimization

Originally PSO is an intelligent simulation of birds foraging behavior. In this process, each individual reference to the information from other members of the group and its own experience to choose the next best foraging sites. Through multiple iterations, each individual can choose a best site. Standard Particle Swarm Optimization (SPSO) is the most basic algorithm of PSO, and it can be described in mathematics as follows:

Assuming that the search space is D-dimensional, and the swarm has N particles. The i^{th} particle is represented by a D-dimensional vector $X_i = (x_{i1}, x_{i2}\ldots x_{iD})$ and the best particle in the swarm is denoted by the index X_g. The speed of the i^{th} particle is defined as the mobile distance in each iteration, represented as $V_i = (v_{i1}, v_{i2}\ldots v_{iD})$ and its best previous position is recorded as $P_i = (p_{i1}, p_{i2}\ldots p_{iD})$. Then the particles move according to the following equations:

$$v_{id} = w * v_{id} + c_1 * r_1(p_{id} - x_{id}) + c_2 * r_2(p_{gd} - x_{id}) \quad (1)$$

$$v_{id} = \begin{cases} v_{max}, if\ v_{id} \geq v_{max} \\ -v_{max}, if\ v_{id} \leq -v_{max} \end{cases} \quad (2)$$

where i = 1, 2, 3 ….., N;

d = 1, 2, 3, …., D;

w is the inertia weight;

c_1 and c_2 are two positive constants;
r_1 and r_2 are two random numbers in the range [0-1]
v_{max} is used to limit the particle speed to improve the running effect of the algorithm.

3 KDDCUP'99 DATA SET

The data set used to perform the experiment is taken from KDD Cup'99 [7][8][9], which is widely accepted as a benchmark dataset and referred by many researchers. "10% of KDD Cup'99" from KDD Cup '99 data set was chosen to evaluate rules and testing data sets to detect intrusion. The entire KDD Cup'99 data set contains 42 features. Connections are categorized into five main categories:

1. Normal connections
2. DOS – Denial of Service
3. Probe – e.g. Port Scanning
4. U2R – unauthorized access to root privileges,
5. R2L – unauthorized remote login to machine.

The initial training set, however, contained 22 attack types. Since the objective of our project is not to detect each of these attack types but rather to detect the major categories into which these attacks fall, we aggregated the 22 attack types into the above five generic categories as follows:

• DOS - Back, land, Neptune, pod, smurf, teardrop.
• Probe - Ipsweep, nmap, portsweep, lht_port_attack.
•R2L(Remote-to-Local) - Imap, ftp_write, guess_passwd, multihop, phf, spy,warezclient, warezmaster.
•U2R(User-to-Root) - buffer_overflow, Rootkit, Perl, Loadmodule.

The 42 attributes of a connection are as follows:
duration, protocol_type, service, flag, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, num_failed_logins, logged_in, num_compromised, root_shell, su_attempted, num_root, num_file_creations, num_shells, num_access_files, num_outbound_cmds, is_hot_login, is_guest_login, count, srv_count, serror_rate, srv_serror_rate, rerror_rate, srv_error_rate, same_srv_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_count, dst_host_same_srv_rate, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_srv_serror_rate, dst_host_rerror_rate, dst_host_srv_error_rate and class.

4 GENETIC ALGORITHM

A genetic algorithm (GA) is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution. The algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm randomly selects individuals from the current population and uses them as parents to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution. In our project we are using the Genetic Algorithm for the feature selection purpose. All the network connection will have total 42 attributes, among which we will select 15 best features for the further PSO implementation as it will optimize the PSO with respect to time.

4 ADAPTIVE MUTATION

It is used for omitting the early convergence of PSO. It is implemented on the attribute values of the particles. If the value of any attribute of the particle is below some threshold value, adaptive mutation is done on these values to differ the value from the previous one which effectively improves the performance of PSO.
Thus, the particles are mutated as per the following rule:

$$f(x) = f(x) + \sigma 1 \times \text{betarnd}(1,1)$$
where
$$\sigma 1 = \sigma \times e^{(\tau \times r1 + \tau 1 \times r2)},$$
$$\tau = \frac{1}{\sqrt{2 \times N}} \text{ and } \tau 1 = \frac{1}{\sqrt{2 \times \sqrt{N}}}$$

betarnd(1,1) is a random number generated by beta distribution with parameters less than 1, r1 and r2 are two random numbers and N is total number of attributes associated with each particle.

5 ALGORITHM DESCRIPTION

Step 1: Data Mining

The total number of records present in "10% of KDD Cup'99" dataset is 494021. However, not all data is present in numerical format. There are four attributes which are present as string are assigned distinct numerical values.

Step 2: Feature Selection

Most of the attribute values are zeros or insignificant for a particular connection. Thus, in order to select the most appropriate attributes to carry out PSO to generate a ruleset to detect intrusions, Genetic Algorithm is used. Total 15 attributes are selected out of 42.

Step 3: Finding Boundaries

The maximum and minimum values of each attribute is calculated and stored separately to limit the values of all the attributes during each iteration.

Step 4: Velocity Calculation

Each particle is assigned a velocity which is given by
$v = v_{min} + (v_{max} - v_{min}) * r$
where v_{min} is 0.0
v_{max} is 4.0
and r is a random variable

The velocities are then added to respective attributes of each particle and then again the particle values are limited based on the boundaries calculated in Step 3.

Step 5: Adaptive Mutation

Once the velocities of the particles are calculated, Adaptive Mutation is applied on the pBest values of the particles which are initialized to minus infinity.

Step 6: Fitness Function

The fitness function is used to evaluate the quality of the particles. The quality of a particle with a higher value is better. The fitness values of the particles are then calculated based on the following function:

$$F(X_p) = -\sum_{1}^{N} {X_{pN}}^2$$

where N is the number of attributes of each connection,
p is the particle number and
$X_{pN}$is the value of attribute.
The fitness values are then used to calculate the particle best, pBest values and global best, gBest values as follows:
If fitness value > pBest value, then pBest value = fitness value, andthe best pBest value is assigned to the gBest value. The gBestPosition will contain the values of all 15 attributes corresponding to the particle having the best pBest value.

Step 7: PSO calculation

The velocities are then updated using the PSO equations (1) and (2) and are then limited by maximum and minimum values of the velocity.

Step 8: Step 5, 6 and 7 are iterated several times to get the best optimized values of connection attributes which will facilitate the intrusion detection.

Step 9: Intrusion Detection

The particle values are then compared with ±5% of the gBest values and are categorized as per their attack type. Those particles satisfying this limit are considered as intrusions and are registered separately. The final report is then generated specifying total number of intrusions present in the dataset and total number of intrusions detected and are categorized based on the above four types of intrusions present.

6 Experimental Results

The Particle Swarm Optimization algorithm is being run on 490000 connections of the KDDCup Data Set and the results obtained under four major types of attacks, i.e., DOS, U2R, R2L and PROBE are mentioned in the table 1. The results, thus, obtained are quite satisfactory.

TABLE 1
RESULTS FOR INTRUSION DETECTION

S.No.	Type of Attack	No. of Attacks Present	Percentage of Attacks Detected
1	DOS	390493	98.76
2	U2R	52	78.84
3	R2L	1126	90.76
4	PROBE	4107	91.42

7 Conclusion

This novel method of intrusion detection in a network using Particle Swarm Optimization implemented with the help of KDD Cup '99 data set is found to be effective in detection of the intrusions and feasible. This methodology uses genetic algorithm for feature selection and adaptive mutation for slow convergence of the algorithm. The results, thus, obtained proves that the devised method for intrusion detection is efficient.

References

[1] Kennedy J, Eberhart RC, "Particle Swarm Optimization," In:Proceedings of the IEEE Int. Conf. Neural Networks: 1942–1948, 1995.

[2] Parsopoulos KE, Plagianakos VP, Magoulas GD and Vrahatis MN,"Streching Technique for Obtaining Gloabal Minimizers ThroughParticle Swarm Optimization," In: Proceedings Particle SwarmOptimization Workshop: 22–29, 2001.

[3] Chen Guolong, Chen Qingliang and Guo Wenzhong, "A PSO-BasedApproach to Rule Learning in Network Intrusion Detection," FuzzyInformation and Engineering (ICFIE), ASC 40, pp. 666–673, 2007.

[4] Chebrolu S.et al. Feature deduction and ensemble design of intrusiondetection systems. Computer & Security, 2004, 24(4): 295-307.

[5] Opitz DW. Feature selection for ensembles. In: Proc. of the 16thNationalConf.on Artificial Intelligence(AAAI).Orlando:AAAIPress,1999.379-384.

[6] Tsymbal A.et al. Ensemble feature selection with the simple Bayesian classification. Information Fusion,2003,4(2):87-100.

[7] Ozgur Depren, Murat Topallar, Emin Anarim, M. Kemal Ciliz."An Intelligent Intrusion Detection System (IDS) for anomaly and misuse detection in computer networks". Expert Systemswith Applications 29 (2005) 713–722 Expert Systems withApplications 29 (2005)713722.www.elsevier.com/locate/eswa.

[8] H. Güneş Kayacık, A. Nur Zincir-Heywood, Malcolm I.Heywood, "Selecting Features for Intrusion Detection: ASpector, AFeature Relevance Analysis on KDD 99 Intrusion DetectionDatasets". Dalhousie University, Faculty of Computer Science,http://www.cs. dal. ca /projectx/

[9] The KDD Archive. KDD99 cup dataset, 1999.