A Hybrid Anomaly Detection Model using G-LDA

Bhavesh Kasliwal [a] , Shraey Bhatia [a], Shubham Saini [a] , I.Sumaiya Thaseen [a], Ch.Aswani Kumar [b]
[a,]School of Computing Science and Engineering, VIT University, Chennai, Tamil Nadu,India.
[b] School of Information Technology and Engineering, VIT University, Vellore, Tamil Nadu, India.

[a]sumaiyathaseen@gmail.com

Abstract— **Anomaly detection is one of the important challenges of network security associated today. We present a novel hybrid technique called G-LDA to identify the anomalies in network traffic. We propose a hybrid technique integrating Latent Dirichlet Allocation and genetic algorithm namely the G-LDA process. Furthermore, feature selection plays an important role in identifying the subset of attributes for determining the anomaly packets. The proposed method is evaluated by carrying out experiments on KDDCUP'99 dataset. The experimental results reveal that the hybrid technique has a better accuracy for detecting known and unknown attacks and a low false positive rate.**

Keywords—Anomaly, Latent Dirichlet Allocation, Genetic Algorithm, Breeding, Fitness, Intrusion Detection Systems

I. INTRODUCTION

There has been exponential rise in people being connected in the current era through various networks like LAN, MAN or WANs. With this augment of network traffic there is also an increase in crimes like frauds, attacks and intrusions in networks. Though firewalls, cryptic methods are used for security many systems are vulnerable to attack due to the incompetent nature of the security models. Thus, Intrusion Detection Systems (IDS) are in place to deal with these problems associated with network intrusions. An intrusion is defined as a sequence of related actions performed by a malicious adversary that results in the compromise of a target system [1]. It is assumed that actions of intruder violate a given security policy.

Detection of anomalies is a very active area associated with network as large amounts of data has to be processed. Network traffic has to be characterized by some features such as destination and source address before applying the data to machine learning or evolutionary algorithms. The design and development of IDS [2] involves data collection, data pre processing, intrusion identification and response. Intrusion recognition is very critical in any IDS as in this phase it compares the audit with detection patterns to identify intrusive behavior.

Many intelligent systems for intrusion detection have been developed using an ensemble of soft computing techniques [14]. Integration of many techniques like genetic algorithm, neural network and decision tree [15][16][17] has lead to improved accuracy in the detection and prevention of intrusions on the basis of observed activity. The hybrid nature of different learning techniques aids in overcoming the individual limitation and achieve better results for intrusion detection.

II. RELATED WORK

The literature indicates that authors have employed Latent Dirichlet Allocation [3] (LDA) and Genetic algorithms individually for network modeling and intrusion identification. Some of them are as follows.LDA for traffic analysis was initially employed by Cramer et al [4].Benjamin D. Newton et al [5] applied LDA for anomaly detection on network traces at University of North Carolina at Chapel Hill (UNC). The authors used LDA on packet counts, user sessions, documents and port numbers. Moreover, We Lie et al [6] used genetic algorithms for intrusion detection. Zhang et al [7] employed LDA to provide an accurate analysis of whether the network traffic model is of actual traffic type. Using the linear discriminant arithmetic, data sets generated by a network simulator were analyzed for identification of complex network traffic. Gomathy [8] et al proposed feature selection by employing genetic algorithm for better accuracy and efficiency. The authors also integrated back propagation neural network to evaluate the performance of detector based on detection accuracy. Siva et al [9] proposed a light weight intrusion detection system to identify network anomalies. The authors constructed a wrapper based feature selection algorithm and integrated with an ensemble of neural decision trees to achieve better detection accuracy. The genetic algorithm was used for feature extraction. Kavitha et al [10] proposed an emerging approach for intrusion detection system using a combination of fuzzy logic, intuitionistic logic, paraconsistent logic, and three valued logics. An improvised genetic algorithm was applied on generated rules for validating data set. Sumaiya et al[18] gave an analysis of supervised tree based classifiers for intrusion detection system wherein different classifier models along with feature selection was applied to obtain an optimized record set for determining whether a packet is of normal or anomaly type.

The G-LDA technique was previously introduced in the work "Spam Detection using G-LDA" [11]. The authors employed LDA using Gibbs Sampling over the Enron Spam Corpus. LDA is a technique for modeling sparse vectors of count data

978-1-4799-2572-8/14/$31.00 ©2014 IEEE

such as bags of features or bags or words. LDA assumes that words in each document belong to a topic, where each topic is represented as a multinomial probability distribution over words. LDA can be applied to diverse domains and not just Natural Language Processing, which it is renowned for. It can be used for generating a model of the network data to identify anomalous traffic. The results of the LDA are subjected to genetic evolution. This involves evaluating the fitness of the items and breeding them through generations. Fitness function is used to evaluate the figure of merit of an item. Items (parents) picked using the roulette wheel method are bred using the OR method. After each round of breeding, m new items (children) are produced and n worst items are deleted. The advantages of using a genetic algorithm are as follows: i) Anomalies do not follow a fixed protocol and since genetic algorithms follow evolutionary techniques, the model may be able to detect unknown anomalies. Ii) Genetic algorithm uses Darwin's survival of fittest theory so that the system can detect prominent anomalies in any domain.

The significance of the proposed network intrusion model using G-LDA is as follows: i) an analysis of intrusion detection using LDA and genetic algorithm is not attempted before and it is a novel technique for identifying the intrusions in the network as it produces best results in an combined environment. ii) The previous work proposed by We Lie et al [6] assigned weight in the genetic module for every attribute whereas in the proposed model we deal with weight for a network packet as whole i.e. same weight for all attributes. The proposed method has been analyzed on KDD data set with huge sized sample containing 100,000 network packets

III. PROPOSED WORK

Figure 1 and Figure 2 gives the overview of the proposed system. The anomaly and normal class packets from the KDDCUP'99 [12] set were taken separately for training. Preprocessing involving the removal of header data, and converting into the format required for applying LDA was done. Set generation using LDA and attribute selection was performed in parallel and output given to the Genetic Algorithm. The Genetic algorithm included computing initial score of the data items, followed by breeding, fitness evaluation and filtering to produce a new generation. This led to the construction of a rule set for determining the nature of an incoming packet.

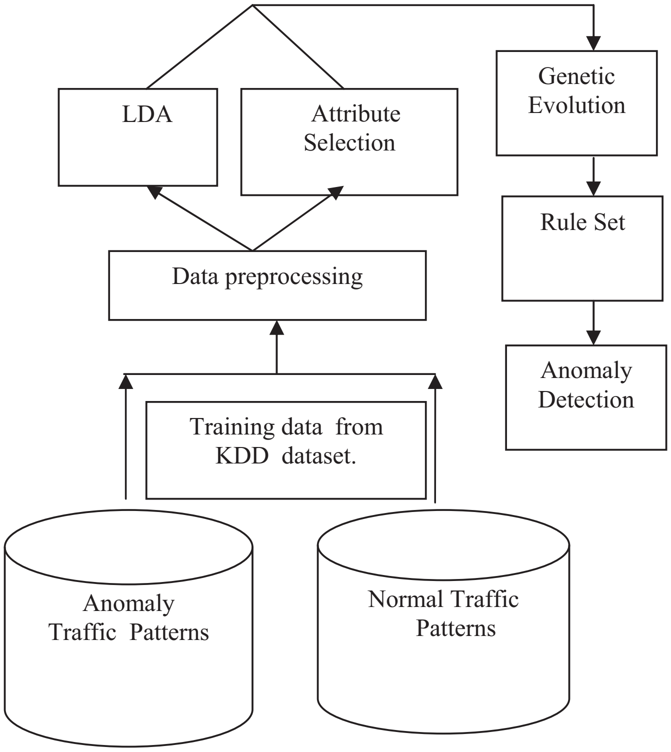

Figure 1. Architecture of G-LDA Process

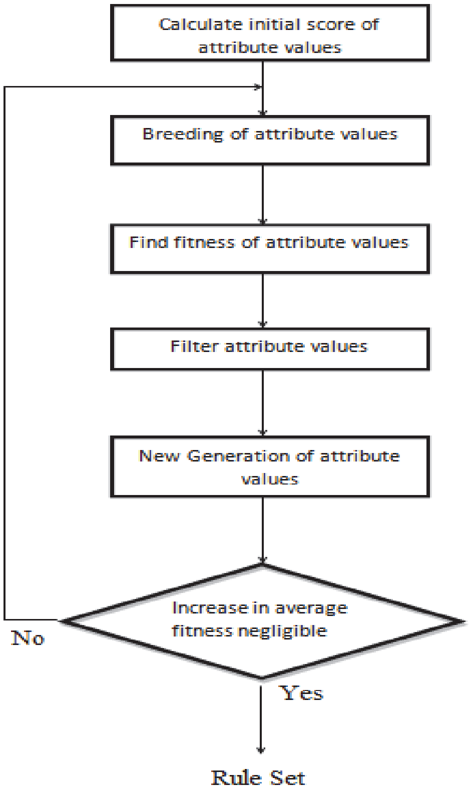
Calculate initial score of attribute values
Breeding of attribute values
Find fitness of attribute values
Filter attribute values
New Generation of attribute values
Increase in average fitness negligible
No
Yes
Rule Set

Figure 2. GA Component

A. Attribute Selection

There are certain attributes in a network packet that play a major role in indentifying the nature of the packet such as normal or anomaly. The mean and mode values of each numerical attribute for both anomaly and normal packets are calculated as this measure is used to identify the best feature subset. The attributes having different mode values for the anomaly and normal packets with their mean close to their mode value were chosen for anomaly detection purpose. Tables I and II show the mode and mean values for the attributes and table III show the best feature subset identified from tables I and II.

TABLE I. MODE VALUES FOR ALL NUMERIC ATTRIBUTES

Attribute	**Anomaly Mode**	**Normal Mode**
duration'	0	0
'src_bytes'	0	0
'dst_bytes'	0	0
'land'	0	0
'wrong_fragment'	0	0
'urgent'	0	0
'hot'	0	0
'num_failed_logins'	0	0
'logged_in'	0	1
'num_compromised'	0	0
'root_shell'	0	0
'su_attempted'	0	0
'num_root'	0	0
'num_file_creations'	0	0
'num_shells'	0	0
'num_access_files'	0	0
'num_outbound_cmds'	0	0
'is_host_login'	0	0
'is_guest_login'	0	0
'count'	1	1
'srv_count'	1	1
'serror_rate'	1	0
'srv_serror_rate'	1	0
'rerror_rate'	0	0
'srv_rerror_rate'	0	0
'same_srv_rate'	1	1
'diff_srv_rate'	0.06	0
'srv_diff_host_rate'	0	0
'dst_host_count'	255	255
'dst_host_srv_count'	1	255
'dst_host_same_srv_rate'	1	1
'dst_host_diff_srv_rate'	0.07	0
'dst_host_same_src_port_rate'	0	0
'dst_host_srv_diff_host_rate'	0	0
'dst_host_serror_rate'	1	0
'dst_host_srv_serror_rate'	1	0
'dst_host_rerror_rate'	0	0

TABLE II. MEAN VALUES FOR ALL NUMERIC ATTRIBUTES

Attribute	**Anomaly Mean**	**Normal Mean**
duration'	423.32069	168.5873959
'src_bytes'	w82820.141	13133.27933
'dst_bytes'	37524.482	4329.685223
'land'	0.000307	0.000103945
'wrong_fragment'	0.0487464	0
'urgent'	6.82E-05	0.000148494
'hot'	0.1742623	0.230655005
'num_failed_logins'	0.0010404	0.00138099
'logged_in'	0.0340269	0.710645501
'num_compromised'	0.0175678	0.507075717
'root_shell'	0.0005458	0.002034361
'su_attempted'	1.71E-05	0.002049211
'num_root'	0.0027119	0.562924135
'num_file_creations'	0.0016374	0.02227403
'num_shells'	0.0001876	0.000608823
'num_access_files'	0.0001876	0.007498923
'num_outbound_cmds'	0	0
'is_host_login'	0	1.48E-05
'is_guest_login'	0.0053556	0.012963485
'count'	154.84999	22.51794544
'srv_count'	27.797885	27.68565404
'serror_rate'	0.5958079	0.013440892
'srv_serror_rate'	0.5930718	0.012083364
'rerror_rate'	0.20698	0.044195982
'srv_rerror_rate'	0.2091141	0.044629137

'same_srv_rate'	0.306659	0.969360141
'diff_srv_rate'	0.1024095	0.028787847
'srv_diff_host_rate'	0.064079	0.126263309
'dst_host_count'	222.02526	147.4319231
'dst_host_srv_count'	29.929081	190.285761
'dst_host_same_srv_rate'	0.1874174	0.811875028
'dst_host_diff_srv_rate'	0.1321313	0.040133941
'dst_host_same_src_port_rate'	0.1789928	0.121725792
'dst_host_srv_diff_host_rate'	0.0400621	0.025995723
'dst_host_serror_rate'	0.5951772	0.01393003
'dst_host_srv_serror_rate'	0.5913292	0.006116449
'dst_host_rerror_rate'	0.2018103	0.046589252

TABLE III. SELECTED ATTRIBUTES

S.no	Feature
1	logged_in
2	Serror_rate
3	srv_serror_rate
4	Same_srv_rate
5	diff_srv_rate
6	dst_host_serror_rate
7	dst_host_srv_serror_rate

B. Set Generation

In this paper, the KDDCUP '99 dataset is used which is based on the 1998 DARPA. The KDDCUP'99 is the most widely used data set for research in KDD and data mining. We employ this data set for construction and evaluation of our anomaly based model. After converting into a required format, KDD data set is supplied to LDA using JGibbLDA [13] package. JGibbLDA is a Java implementation of LDA using Gibbs Sampling technique. A total of 200 sets with 10 words each were generated from the anomaly packets. Each set contained a single packet type as majority. Although there were not 200 different packet types in the data set, it was found that distributing the packets over a large number of sets gave the best results. The similar process was applied to the normal packets also to obtain unique sets.

C. Genetic Algorithm

The attribute subset is retrieved from each of the 2000 generated packets separately and genetic evolution is applied on every attribute. Table IV is the step wise process involved in the GA.

TABLE IV. STEP WISE PROCESS INVOLDED IN GA

Steps	Process
1	Loop till Increase in average fitness negligible
2	Perform breeding of attribute values using roulette wheel selection
3	Find fitness of attribute values
4	Filter attribute values to produce a new generation
5	To calculate the fitness value. Fitness = Number of packets*(1+ $\frac{N-N_w}{N}$) Where N = Threshold value for restricting the number of generations (Taken it as 2) N_w = No. of values in the term

Threshold value produces a negative weight. Any term longer than N numbers will receive fitness penalty, and keywords less than N numbers receive a bonus.

After applying the genetic algorithm for 3 generations, the top three fittest unique values are obtained from each of the fields. The frequency of those 3 attribute values are also calculated based on the training data set. The process is done for both anomaly and normal data sets. Tables V and VI shows the highest three fittest values of the attributes after applying genetic evolution over 3 generations.

TABLE V. HIGHEST THREE FITNESS VALUES OF ANOMALY ATTRIBUTE SUBSET AFTER APPLYING GENETIC EVOLUTION OVER 3 GENERATIONS

Attribute name	Anomaly top 3 values	Frequency of anomaly top 3
Logged in	0 1	48284 1711
Serror_rate	1 0 0.01	29020 18492 181
Srvr_serror_rate	1 0 0.43	29446 20242 2
Diff_srv_rate	0.06 0 0.05	16167 11441 5783
Dst_host_diff_srv_rate	0.07 0 0.05	13419 5892 4974
Dst_host_serror_rate	1	28573

	0 0.01	17092 470
Dst_host_srv_serror_rate	1 0 0.43	29446 20242 2

TABLE VI. HIGHEST THREE FITNESS VALUES OF NORMAL ATTRIBUTE SUBSET AFTER APPLYING GENETIC EVOLUTION OVER 3 GENERATIONS

Attribute name	**Normal top 3 values**	**Frequency of normal top 3**
Logged in	0 1	14503 35497
Serror_rate	0 1 0.5	48376 310 346
Srvr_serror_rate	0 0.04 0.5	48294 58 263
Diff_srv_rate	0 0.01 1	46578 747 755
Dst_host_diff_srv_rate	0 0.01 0.07	29775 6323 822
Dst_host_serror_rate	0 0.01 0.02	45543 2068 575
Dst_host_srv_serror_rate	0 0.04 0.5	48294 58 263

D. *Determing the type of incoming packet*

For determining if an incoming packet is anomaly or not, the following procedure is performed:

```
• For each selected attribute value $F_i$ in incoming packet
      If $F_i \in V_i$
            $S_i$ = (A* Frequency of $F_i$ in Anomaly) – (Frequency of $F_i$ in Normal)
      Else
            $S_i = 0$
• $C = \sum S_i$
• If $C > 0$
      o Then Anomaly
• Else Normal
```

Here A is the additional weight that is multiplied to the anomaly frequency.

An additional weight is required since the algorithm detects generic anomalies having diverse values for each of the fields unlike the normal packets that contain values in a particular range.

IV. RESULTS

The experiments were conducted on the Intel i5 processor with 4GB memory in Windows environment. All the programming was done using Java SDK 7. The output from the Set Generation stage were imported into a MySQL database, which were then processed via Java using the JDBC interface. For testing the efficiency of the rule set, 50000 Anomaly and 50000 Normal class packets from the KDDCUP'99 set were used

Table VII shows the number of sample instances taken for training and test data set. We measure the performance of our model using the following metrics. These values are true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

i) Accuracy (Acc): (TN+TP)/(TN+TP+FN+FP);
Proportion of the total number of predictions that were correct.

ii) Precision Rate (PR): TP/(TP+FP)
Proportion of the predictive positive cases that were correct.

iii) Detection Rate (DR) : TP/(TP+FN)
Number of intrusion samples detected by the model (True Positive) divided by the total number of intrusion samples present in the test set.

iv) False Positive Rate (FPR): FP/(TN+FP)
Number of samples misclassified as anomalies.

Tables VIII and IX summarizes the performance metrics for various additional weights assigned to the anomaly frequency to determine the optimized accuracy for the proposed model. The aim is to attain maximum possible accuracy, with least false positive rate. As the value of weight increases, both the accuracy and the false positive rate increases gradually, thus requiring a need for a trade-off between the accuracy and the false positive rate. We conclude that when the weight is 1.75, we obtain an accuracy of 0.885 with a false positive rate of 0.06. Hence this weight measure can be fixed for identifying whether an incoming packet is of normal or abnormal type.

TABLE VII. NUMBER OF SAMPLE INSTANCES TAKEN FROM KDD DATA SET

	Training Data	Test Data
Anomaly	2000	50000
Normal	2000	50000

TABLE VIII. SUMMARY OF THE PERFORMANCE METRICS COMPUTED OVER DIFFERENT VALUES OF A

Weight	TP	TN	FP	FN
1	60	99.75	0.25	40
1.5	64	99	1	36
1.7	80	95	5	20
1.75	83	94	6	17
1.8	86	91	9	14
1.85	88	78	22	12
2	96.5	70	30	3.5

TABLE IX. SUMMARY OF THE PERFORMANCE METRICS COMPUTED OVER DIFFERENT VALUES OF A

Weight	DR	PR	Acc	FPR	F-score
1	0.6	0.995851	0.79875	0.0025	0.74883
1.5	0.64	0.984615	0.815	0.01	0.775758
1.7	0.8	0.941176	0.875	0.05	0.864865
1.75	0.83	0.932584	0.885	0.06	0.878307
1.8	0.86	0.905263	0.885	0.09	0.882051
1.85	0.88	0.8	0.83	0.22	0.838095
2	0.965	0.762846	0.8325	0.3	0.852097

V. CONCLUSION

We propose a novel hybrid technique for detecting anomalous network traffic using G-LDA. Latent Dirichlet Allocation has been proved useful for generating a model of the network data to identify anomalies in network traffic. Anomaly detection was done by choosing an attribute subset of network traffic. As network attacks become more sophisticated and unpredictable continuously, performing Genetic Evolution techniques to the modeled network data allows us to detect previously unknown anomalous network data.

We evaluated the performance of our proposed system. The G-LDA process for detecting anomalous network traffic showed an accuracy of 88.5% with a 6% false positive Rate.

It may be noted that the process was applied on a generic anomaly data. Applying the same over a specific anomaly type may lead to an increased accuracy, and is intended as future work.

REFERENCES

[1] Valeur, Fredrik, and Giovanni Vigna. Intrusion detection and correlation: challenges and solutions. Vol. 14. Springer, 2005.

[2] Kim, Dong Seong, and Jong Sou Park. "Network-based intrusion detection with support vector machines." Information Networking. Springer Berlin Heidelberg, 2003.

[3] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research,Volume 3, pp.993-1022,2003.

[4] Cramer, Christopher, and Lawrence Carin. "Bayesian topic models for describing computer network behaviors." Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on. IEEE, 2011.

[5] Newton, Benjamin D. "Anomaly Detection in Network Traffic Traces Using Latent Dirichlet Allocation."

[6] Li, Wei. "Using genetic algorithm for network intrusion detection." Proceedings of the United States Department of Energy Cyber Security Group,pp1-8,2004.

[7] Bing-Yi Zhang,Ya-Min Sun,Yu-Lan,Bian,Hong Ke Zhang,"Linear Discriminant Analysis in network traffic modeling", International Journal of Communication Systems",Volume 19,Issue 1,pp.53-65,2006.

[8] A.Gomathy and B.Lakshmi,"Network intrusion detection using Genetic algorithm and Neural Network", Communications in Computer and Information Science,Volume 198,pp.399-408,2011.

[9] Siva S,Sivatha Sindhu,S.Geetha,A.Kannan,"Decision tree based light weight intrusion detection using a wrapper approach",Expert Systems with applications,Volume 39,pp.129-141,2012.

[10] B.Kavitha,S.Karthikeyan,P.Sheeba Maybell,"An ensemble design of intrusion detection system for handling uncertainity using neutrosophic logicclassifier",Knowledge based systems,Volume 28,pp.88-96,2012.

[11] Saini, Shubham, Bhavesh Kasliwal, and Shraey Bhatia. "Spam Detection using G-LDA." International Journal of Advanced Research in Computer Science and Software Engineering,Volume 3,Issue 10,pp.406-409,2013.

[12] Cup, K. D. D. "Available on: http://kdd. ics. uci. edu/databases/kddcup 99/kddcup99. html.",2007.

[13] Phan, Xuan-Hieu, and Cam-Tu Nguyen. "Jgibblda: A java implementation of latent Dirichlet allocation (lda) using gibbs sampling for parameter estimation and inference",2006.

[14] Shekhar R Gaddam, Vir V Phoha and Kiran S Balagani,"A novel method for supervised anomaly detection by cascading K-Means clustering and ID3 deicsion tree learning methods", IEEE transactions on knowledge and data engineering,Volume.19,pp.345-354,2007.

[15] Amor, Nahla Ben, Salem Benferhat, and Zied Elouedi. "Naive Bayes vs decision trees in intrusion detection systems" Proceedings of the 2004 ACM symposium on Applied computing, pp.420-424,2004.

[16] Benferhat, S. and Tabia, K., "On the combination of Naive Bayes and decision trees for intrusion detection", International Conference on Intelligent Agents, Web Technologies and Internet Commerce,Volume 1, pp. 211–216,2006.

[17] Xiang, C., and Lim, S. M, "Design of multiple-level hybrid classifier for intrusion detection system", IEEE Transaction on System, Man and Cybernetics, Part A: Cybernetics, Volume 2, pp.117–122,2005.

[18] Sumaiya Thaseen and Ch. Aswani Kumar, "An Analysis of supervised tree based classifiers for intrusion detection system", IEEE International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME), February 2013.