Network Intrusion Detection using Supervised Machine Learning Technique with Feature Selection

[1]GUNREDDY LAHARI, [2]Dr,D. William Albert
[1]Research Scholar, [2]Professor
Department of CSE
Ashoka Womens engineering college, Kurnool, AP, India

ABSTRACT

In this paper author is evaluating performance of two supervised machine learning algorithms such as SVM (Support Vector Machine) and ANN (Artificial Neural Networks). Machine learning algorithms will be used to detect whether request data contains normal or attack (anomaly) signatures. Now-a-days all services are available on internet and malicious users can attack client or server machines through this internet and to avoid such attack request IDS (Network Intrusion Detection System) will be used, IDS will monitor request data and then check if its contains normal or attack signatures, if contains attack signatures then request will be dropped.

IDS will be trained with all possible attacks signatures with machine learning algorithms and then generate train model, whenever new request signatures arrived then this model applied on new request to determine whether it contains normal or attack signatures. In this paper we are evaluating performance of two machine learning algorithms such as SVM and ANN and through experiment we conclude that ANN outperform existing SVM in terms of accuracy.

To avoid all attacks IDS systems has developed which process each incoming request to detect such attacks and if request is coming from genuine users then only it will forward to server for processing, if request contains attack signatures then IDS will drop that request and log such request data into dataset for future detection purpose.

To detect such attacks IDS will be prior train with all possible attacks signatures coming from malicious user's request and then generate a training model. Upon receiving new request IDS will apply that request on that train model to predict it class whether request belongs to normal class or attack class. To train such models and prediction various data mining classification or prediction algorithms will be used. In this paper author is evaluating performance of SVM and ANN. In this algorithms author has applied Correlation Based and Chi-Square Based feature selection algorithms to reduce dataset size, this feature selection algorithms removed irrelevant data from dataset and then used model with important features, due to this features selection algorithms dataset size will reduce and accuracy of prediction will increase.To conduct experiment author has used NSL KDD Dataset and below is

some example records of that dataset which contains request signatures. I have also used same dataset and this dataset is available inside 'dataset' folder.

I. INTRODUCTION

With the wide spreading usages of internet and increases in access to online contents, cybercrime is also happening at an increasing rate [1-2]. Intrusion detection is the first step to prevent security attack. Hence the security solutions such as Firewall, Intrusion Detection System (IDS), Unified Threat Modeling (UTM) and Intrusion Prevention System (IPS) are getting much attention in studies. IDS detects attacks from a variety of systems and network sources by collecting information and then analyzes the information for possible security breaches [3]. The network based IDS analyzes the data packets that travel over a network and this analysis are carried out in two ways. Till today anomaly based detection is far behind than the detection that works based on signature and hence anomaly based detection still remains a major area for research [4-5]. The challenges with anomaly based intrusion detection are that it needs to deal with novel attack for which there is no prior knowledge to identify the anomaly. Hence the system somehow needs to have the intelligence to segregate which traffic is harmless and which one is malicious or anomalous and for that machine learning techniques are being explored by the researchers over the last few years [6]. IDS however is not an answer to all security related problems. For example, IDS cannot compensate weak identification and authentication mechanisms or if there is a weakness in the network protocols.

Studying the field of intrusion detection first started in 1980 and the first such model was published in 1987 [7]. For the last few decades, though huge commercial investments and substantial research were done, intrusion detection technology is still immature and hence not effective [7]. While network IDS that works based on signature have seen commercial success and widespread adoption by the technology based organization throughout the globe, anomaly based network IDS have not gained success in the same scale. Due to that reason in the field of IDS, currently anomaly based detection is a major focus area of research and development [8]. And before going to any wide scale deployment of anomaly based intrusion detection system, key issues remain to be solved [8]. But the literature today is limited when it comes to compare on how intrusion detection performs when using supervised machine learning techniques [9]. To protect target systems and networks against malicious activities anomaly-based network IDS is a valuable technology. Despite the variety of anomaly-based network intrusion detection techniques described in the literature in recent years [8], anomaly detection functionalities enabled security tools are just beginning to appear, and some important problems remain to be solved. Several anomaly based techniques have been proposed including Linear Regression, Support Vector Machines (SVM), Genetic Algorithm, Gaussian mixture model, knearest neighbor algorithm, Naive Bayes classifier, Decision Tree [3,5]. Among them the most widely used learning algorithm is SVM as it has already established itself on different types of problem [10]. One major issue on anomaly based detection is though all these proposed techniques can detect novel attacks but they all suffer a high false alarm rate in general. The cause behind is the complexity of generating profiles of practical normal behavior by learning from the training data sets [11]. Today Artificial Neural Network (ANN) are often trained by the back propagation algorithm, which had been around since 1970 as the reverse mode of automatic differentiation [12].

The major challenges in evaluating performance of network IDS is the unavailability of a comprehensive network based data set [13]. Most of the proposed anomaly based techniques found in the literature were evaluated using KDD CUP 99 dataset [14]. In this paper we used SVM and ANN – two machine learning techniques, on NSLKDD [15] which is a popular benchmark dataset for network intrusion.

II. EXISTING SYSTEM

As there is no staff available in unmanned restaurants, it is difficult for the restaurant management to estimate how the concept and the food is experienced by the customers. Existing

rating systems, such as Google and TripAdvisor, only partially solve this problem, as they only cover a part of the customer’s opinions. These rating systems are only used by a subset of the customers who rate the restaurant on independent rating platforms on their own initiative. This applies mainly to customers who experience their visit as very positive or negative.

III. PROPOSED SYSTEM

In order to solve the above problem, all customers must be motivated to give a rating. This paper introduces an approach for a restaurant rating system that asks every customer for a rating after their visit to increase the number of ratings as much as possible. This system can be used unmanned restaurants; the scoring system is based on facial expression detection using pretrained convolutional neural network (CNN) models. It allows the customer to rate the food by taking or capturing a picture of his face that reflects the corresponding feelings. Compared to text-based rating system, there is much less information and no individual experience reports collected. However, this simple fast and playful rating system should give a wider range of opinions about the experiences of the customers with the restaurant concept.

IV. SOFTWARE USED

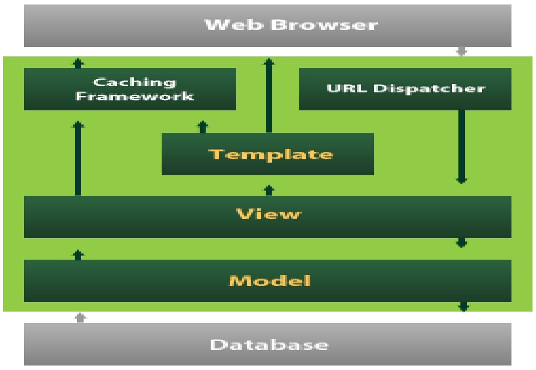
PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library

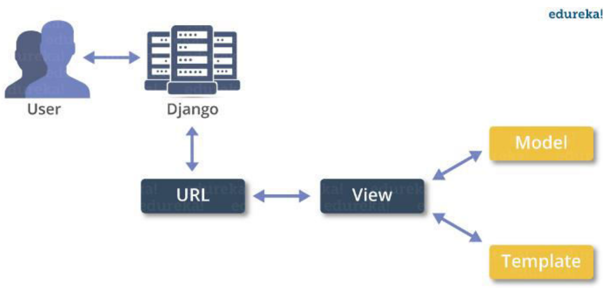
DJANGO

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusabilityand "pluggability" of components, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models.



Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models



V. IMPLEMENTATION

Dataset example

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_host_login,is_guest_login,count,srv_count,serror_rate,srv_serror_rate,rerror_rate,srv_rerror_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_rerror_rate,label

All above comma separated names in bold format are the names of request signature

0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,150,25,0.17,0.03,0.17,0,0,0,0.05,0,normal

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1,1,0,0,0.05,0.06,0,255,9,0.04,0.05,0,0,1,1,0,0,anamoly

Above two records are the signature values and last value contains class label such as normal request signature or attack signature. In second record 'Neptune' is a name of attack. Similarly in dataset you can find nearly 30 different names of attacks.

In above dataset records we can see some values are in string format such as tcp, ftp_data and these values are not important for prediction and these values will be remove out by applying PREPROCESSING Concept. All attack names will not be identified by algorithm if it's given in string format so we need to assign numeric value for each attack. All this will be done in PREPROCESS steps and then new file will be generated called 'clean.txt' which will use to generate training model.

In below line i am assigning numeric id to each attack

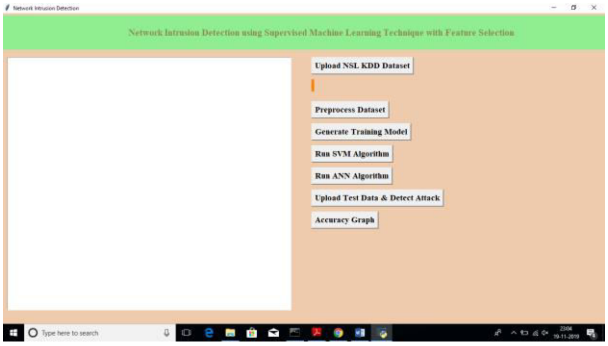
"normal":0,"anamoly":1

In above lines we can see normal is having id 0 and Anomaly has id 1 and goes on for all attacks.

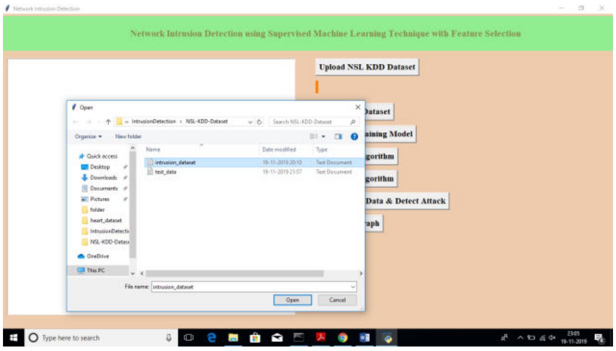
Before running code execute below two commands

VI. RESULTS

Double click on 'run.bat' file to get below screen



In above screen click on 'Upload NSL KDD Dataset' button and upload dataset

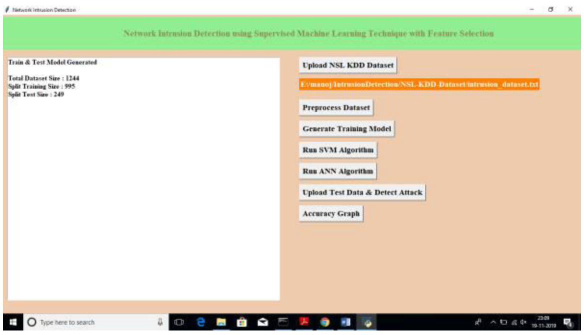


In above screen I am uploading 'intrusion_dataset.txt' file, after uploading dataset will get below screen

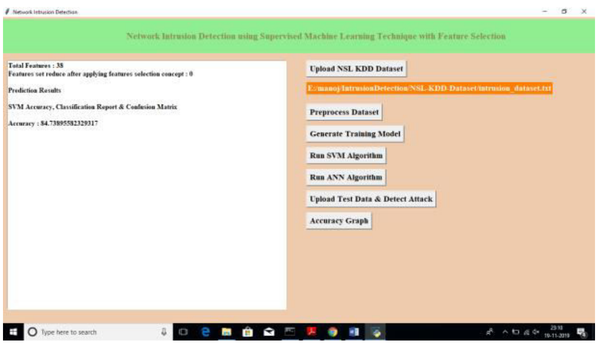
Now click on 'Pre-process Dataset' button to clean dataset to remove string values from dataset and to convert attack names to numeric values

After pre-processing all string values removed and convert string attack names to numeric values such as normal signature contains id 0 and anomaly attack contains signature id 1.

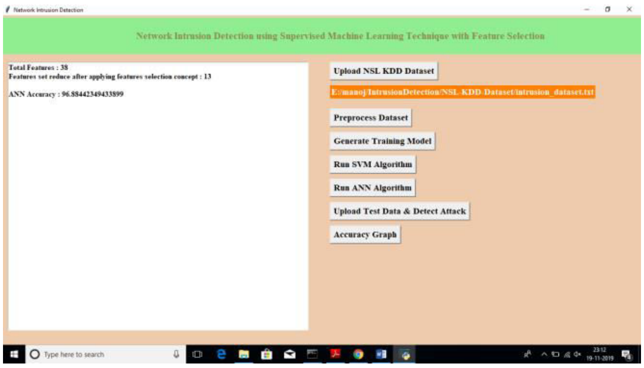
Now click on ‘Generate Training Model’ to split train and test data to generate model for prediction using SVM and ANN



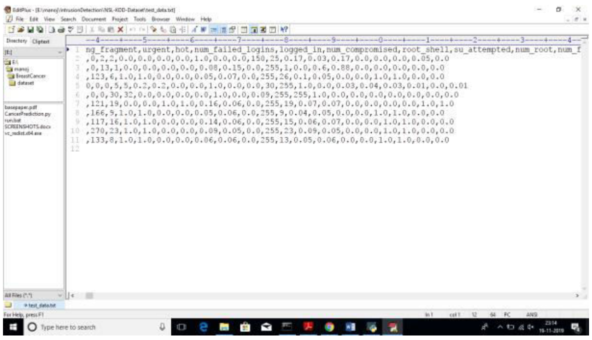
In above screen we can see dataset contains total 1244 records and 995 used for training and 249 used for testing. Now click on ‘Run SVM Algorithm’ to generate SVM model and calculate its model accuracy



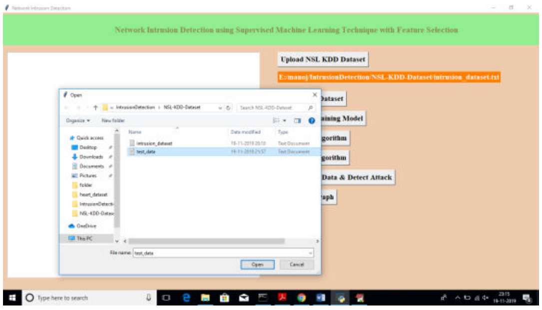
In above screen we can see with SVM we got 84.73% accuracy, now click on ‘Run ANN Algorithm’ to calculate ANN accuracy



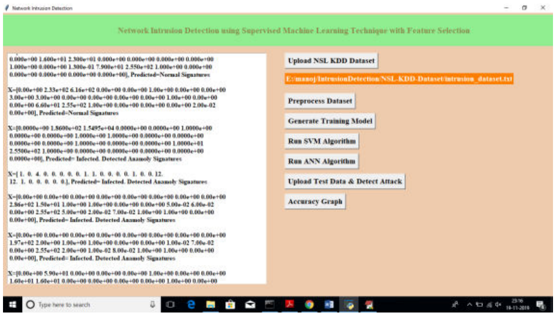
In above screen we got 96.88% accuracy, now we will click on ‘Upload Test Data & Detect Attack’ button to upload test data and to predict whether test data is normal or contains attack. All test data has no class either 0 or 1 and application will predict and give us result. See below some records from test data



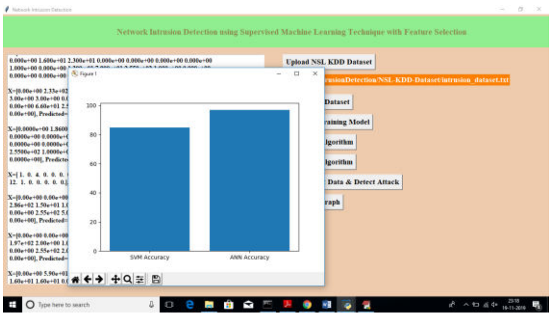
In above test data we don’t have either ‘0’ or ‘1’ and application will detect and give us result



In above screen I am uploading ‘test_data’ file which contains test record, after prediction will get below results



In above screen for each test data we got predicted results as ‘Normal Signatures’ or ‘infected’ record for each test record. Now click on ‘Accuracy Graph’ button to see SVM and ANN accuracy comparison in graph format



From above graph we can see ANN got better accuracy compare to SVM, in above graph x-axis contains algorithm name and y-axis represents accuracy of that algorithms

VII. CONCLUSION

By this project we made a innvoaticve approach for detecting the network intrusion

V. REFERENCES

[1] P. Alaei and F. Noorbehbahani, "Incremental anomaly-based intrusion detection system using limited labeled data," 2017 3th International Conference on Web Research (ICWR), Tehran, Iran, 2017, pp. 178-184.

[2] M. Tavallaee, N. Stakhanova and A. A. Ghorbani, "Toward Credible Evaluation of Anomaly-Based Intrusion-Detection Methods," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 40, no. 5, pp. 516-524, Sept. 2010.

[3] F. Gharibian and A. A. Ghorbani, "Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection," Fifth Annual Conference on Communication Networks and Services Research (CNSR '07), Fredericton, NB, Canada, 2007, pp. 350-358.

[4]N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia,

2015, pp. 1-6.
[5]T. Janarthanan and S. Zargari, "Feature selection in UNSW-NB15 and KDDCUP'99 datasets," 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE), Edinburgh, UK, 2017, pp. 1881-1886.
[6] M. Panda, A. Abraham and M. R. Patra, "Discriminative multinomial Naïve Bayes for network intrusion detection," 2010 Sixth International Conference on Information Assurance and Security, Atlanta, GA, USA, 2010, pp. 5-10.

[7] B. Ingre and A. Yadav, "Performance analysis of NSL-KDD dataset using ANN," 2015 International Conference on Signal Processing and Communication Engineering Systems, Guntur, India, 2015, pp. 92-96.