Intrusion Detection using Machine Learning

GOPINATH ACHUTHAN
Department of Computer Science
Univeristy of Maryland, Baltimore County
Baltimore, MD, USA
rx44815@umbc.edu

***Abstract*— The increase in popularity of internet is raising the risk of network attacks. Network security has become an important issue in our daily life. One way to handle a threat is to generate an alert whenever there is a chance of intrusion and make sure that the alert generated is valid. This paper handles this task by using three different Machine Learning models to develop an Intrusion Detection System that classifies whether an alert generated by the system is suspicious or not. It takes the Investigation Alerts dataset and train these models using semi-supervised learning methods. Once trained, the system is able to classify a new threat as either suspicious or not suspicious.**

***Keywords*— suspicious, recognition, alert, machine learning model, intrusion detection.**

I. Introduction

The importance of privacy is creating a boundary which is protecting us from others. But there are many adversaries outside who tries to bleach the system network. To protect our-self from these attackers, we need an intrusion detection system. The Internet has become a part of daily life and an essential tool today. It aids people in many areas, such as business, entertainment and education, etc. The activities in internet such as email, social network application, banking holds the sensitive information about the customer and providers. Therefore, information security of using Internet needs to be carefully concerned. Intrusion detection is one major research problem for network communication.

As there are many risks of network attacks under the Internet usage, here comes the significant important of the network detection system. The goal of the intrusion detection system is to build a firewall and detect the network attack in the communication or detect the anomaly in the network behavior. Since the growth of attack in network is increasing, the need of intrusion detection system is necessary.

II. BACKGROUND

Intrusion Detection is a classic Machine Learning problem as the output value is discrete. i.e. Threat type. It can be solved through supervised machine learning method. In supervised learning, the system is provided with the correct input and output data, so it learns through it. In other words, we train our system for correct output again and again until it learns a pattern and recognize the output for a new entry. Intrusion Detection Systems are usually categorized into two forms. i.e. Misuse Signature Detection and Anomaly Detection which relies on different approaches and how they are used. When we detect an anomaly, we find out the intrusions which are deviated from the usual patterns and how they are flagged. Whereas, detecting misuse takes into consideration, the working and use of various attack and weak point patterns to identify intrusions.

III. APPROACHES

For such systems, we can use various machine learning approaches where we narrow down the task, use a dataset and apply supervised learning methods to classify the intrusions accordingly. Now a days, neural networks are in demand, so many studies revolves around using these advanced techniques to achieve higher accuracy results. Other classifiers like Support Vector Machines are also very accurate for such problems. If we apply multiple techniques at once, we are basically working on a hybrid approach. Though many solutions have been proposed, the research on this task is still on-going.

Here are some of the machine learning approaches we can consider for such tasks:

1. K-nearest neighbor
2. Support vector machines

3. Artificial neural networks
4. Random forest classifier
5. Self-organizing maps
6. Decision trees
7. Naïve Bayes networks
8. Genetic algorithms
9. Fuzzy logic
10. Hybrid classifiers
11. Ensemble classifiers

Using these classifiers, we can detect whether the incoming alert generated is suspicious or not and act accordingly. For this paper, we chose the three underlined approaches to classify the Investigation dataset. i.e. Support vector machines, Artificial neural networks and Random forest classifier. For classification, we train our model with our dataset of previous alert records. There are different approaches used for this task. We split the training data into training and testing split. The purpose being, we train the model with the training data, and use the testing part of split for classification using SVM classifier, Neural Networks and Random Forest Classifier.

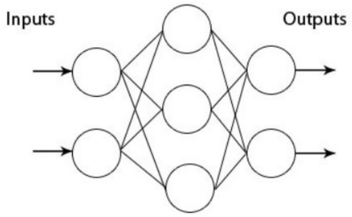
Classification is used in many similar cases now a days. E.g. Email SPAM detection, suicide detection. Text classification is a technique that allows the system to differentiate between text files by classifying them into different classes/categories.

In classification, we categorize a new threat and label it as either.er suspicious or not. This is a binary classification example since the output is fixed and set to 2. For this, we need to have knowledge about the past observations and decide on how to mark a random text as Attack or not. For this purpose, the dataset is already marked with valid labels for each threat which might be one short sentence or a longer paragraph. Since we have divided our dataset into two portions.

We use the training portion to train our machine learning model and use the testing split with new set of threat text, where our classifier tries to predict the label with the higher probability against each threat.

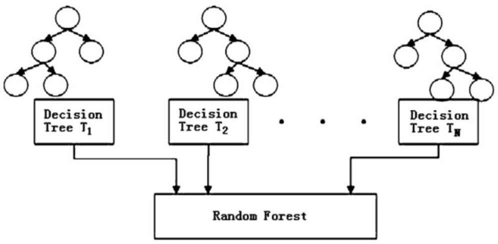
For this approach, various classifiers are used as mentioned above. Neural Networks are one of the most advanced approaches for this task, the reason being high accuracy results but at the same time, they require a lot of time to process the data, since they use advanced mathematical equations and distributions

Each of these approaches gives different results, which will be shared in the Results section of this paper.



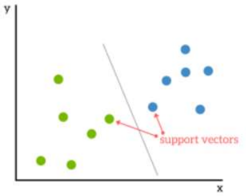
Neural Network

On the other hand, random forest classification algorithm classifies output using bunch of decisions trees. Each tree creates uncorrelated set of trees and when they decide together, they generate better output as compared to a single decision tree. We can use Random Forest algorithm for regression tasks as well. i.e. tasks when our output value is not discrete, instead a distribution or very large set of values which never ends.



Random Forest Classifier model

Finally, one of the most used and famous classifiers is Support Vector Machine, which has been extensively used by researcher in the past due to its high precision and results during classification. It uses hyperplane to disambiguate between correct and incorrect output. i.e. Threat as either Attack threat or Not-Attack threat. The hyperplane with the highest margin is selected as the correct output and all data points that lies in that plane are marked as the correct output result. E.g. In this figure, the line drawn by Vector Support Classifier separates two hyperplanes, marks the green dots as ATTACK threat and blue dots as Not-Attack threats.



Support Vector Machines

IV. ANALYSIS AND RESULTS

We are using the dataset available as the csv file which is taken from knowledge pit website under IEEE Big Data 2019 Cup: Suspicious Network Event Recognition competition. The dataset consists of alerts investigated by a SOC team at SoD. Our target column in the dataset is “notified”, this column gives the result whether the client to notified or not regarding the attack. The value zero means not notified and value one means notified to the client. Here client is the user of the intrusion detection system.

We split the dataset in the ration of 7 to 3 for training data and testing data respectively. Here we split it randomly. Then we use the three different machine learning model to analysis the same dataset. Each model trains the training dataset in different way according to that particular model. After training the dataset, we test the testing data, then we obtain the confusion matrix for the testing data, which is used describe the performance of the classifier machine model.

The accuracy of the model is calculated using the formula:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

Where TP – True Positive
TN – True Negative
FP – False Positive
FN – False Negative

The term true positive represent the hit rate of positive out of all positive response of the system. The False positive is opposite of true positive, it represents the hit rate of not positive out of all positive. The True negative represents the hit rate of negative out of all negative and false negative represents the non-negative out of all negative.

The accuracy value for each model is given below

Classifier model	Accuracy
Random Forest Classifier	0.94
SVM Classifier	0.94
Neural Network	0.94

Accuracy Table for each classifier model

As we can see that the accuracy value for each model is same. With the accuracy value alone the performance of the three different classifiers cannot be measured. So, move to the other metrics to measure the performance of the model classifiers.

Some metrics other than accuracy are precision, recall, F1 score. The precision of the model tells about how often the system predicts positive. That is, the percentage of result which all related. The recall gives the percentage of total relevant results correctly classified by the model. Generally, the term F score means the weighted harmonic mean of the precision and recall. Then, F1 score means the weightage value is one for the F score, which is considered as a baseline case for F score. The formula for the precision, recall and F1 score is given below:

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\text{F1 score} = \frac{2*Precision*Recall}{Precision+Recal}$$

Using this metrics, the performance of the model can be determine. The metrics values for the classifier is given in below tables.

Class	Precision	Recall	F1 score
0	0.95	0.99	0.97
1	0.50	0.20	0.28

Results for Random Forest Classifier

Class	Precision	Recall	F1 score
0	0.94	1.00	0.97
1	0.71	0.01	0.01

Results for SVM Classifier

Class	Precision	Recall	F1 score
0	0.95	1.00	0.97
1	0.57	0.09	0.16

Results for Neural Network Classifier

Note: Class 0 represents not notified and Class 1 represents the notified.

From the above data, the value of the metrics for class 0 is high, because of the dataset i.e., we have the huge dataset which is used to train the model compared to the class 1. So that the metrics values for class 1 is low. The SVM classifier gives the best performance in terms of precision. The Random Forest Classifier gives the best performance in terms of recall and F1 score. Here we call see that, the SVM classifier and Neural Network classifier gives the full recall value for class 0, but for class 1 the results are bad.

From analysis the classifier model using these metrics, the overall best performance is shown in Random Forest Classification model.

V. CONCLUSION

Nowadays, the network security becomes necessary in daily life. Due to raise of new attacks, the intrusion detection system has to adapt to the attack by finding the anomaly in the network system. Therefore, the machine learning classification model is used in order to find the attack and save our data from the adversary.

REFERENCES

1. Debar, H., Becker, M., & Siboni, D. (1992). A neural network component for an intrusion detection system. Proceedings 1992 IEEE Computer Society Symposium on Research in Security and Privacy, 240-250.

2. Liao, H. J., Lin, C. H. R., Lin, Y. C., & Tung, K. Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36(1), 16-24.

3. Smaha, S. E. (1988, September). Haystack: An intrusion detection system. In [Proceedings 1988] Fourth Aerospace Computer Security Applications (pp. 37-44). IEEE.

4. Rowland, C. H. (2002). U.S. Patent No. 6,405,318. Washington, DC: U.S. Patent and Trademark Office.

5. Rowland, C. H. (2002). U.S. Patent No. 6,405,318. Washington, DC: U.S. Patent and Trademark Office.

6. Peddabachigari, S., Abraham, A., Grosan, C., & Thomas, J. (2007). Modeling intrusion detection system using hybrid intelligent systems. Journal of network and computer applications, 30(1), 114-132.

7. Heady, R., Luger, G., Maccabe, A., & Servilla, M. (1990). The architecture of a network level intrusion detection system (No. LA-SUB-93-219). Los Alamos National Lab., NM (United States); New Mexico Univ., Albuquerque, NM (United States). Dept. of Computer Science.