JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Detecting Cyber Attack in Network Dataset using Machine Learning

Mr. Lagineni Sai Kiran. Master Of Computer Applications, Madanapalle Institute of Technology and Science, Andhra Pradesh

Abstract - Malicious cyber-attacks can lurk in enormous amounts of legitimate data in unbalanced network traffic. In cyberspace, it uses a high level of stealth and obfuscation, making it difficult for Network Intrusion Detection Systems (NIDS) to ensure detection accuracy and completeness. Computer vision and deep learning are investigated in this paper for malware detection in unbalanced network data. To address the problem of class imbalance, it presents a novel Difficult Set Sampling Technique (DSSTE) algorithm. To begin, partition the imbalanced training set into the challenging and easy sets using the Edited Nearest Neighbor (ENN) algorithm. Then, to minimize the majority, apply the KMeans technique to compress the majority samples in the challenging set. In the challenging set, zoom in and out the continuous properties of the minority samples, then synthesis fresh samples to enhance the minority number. Then, the enhancement data are mixed with the simple set, the reduced set of majorities in the challenging, and the minority in the tough set to create a new training dataset. The method evens out the initial feature set's imbalance and generates tailored data supplementation for the minority group that wants to understand. It allows the classification algorithm to better learn the distinctions in the training process and increase classification accuracy. We test the suggested strategy on the classic infiltration dataset NSL-KDD as well as the more recent comprehensive intrusion sample CSE-CIC-IDS2018. We employ traditional categorization methods such as random forest (RF) and support vector machine (SVM) (SVM), Mini-VGGNet, XGBoost, MLP AlexNet We evaluate our suggested DSSTE algorithms to some other 24 techniques; the test data showed that the proposed method approach outperforms the others.

1. INTRODUCTION

1.1 Introduction

People can now access a variety of useful services thanks to the advancement and enhancement of Internet technology. However, we are also vulnerable to a variety of security dangers. Network infections, surveillance, and malicious attacks are on the rise, leading society and federal departments to pay more attention towards cybersecurity. Thankfully, cyber security can effectively address these issues. In order to ensure network information security, intrusion detection is crucial. However, as the Internet industry grows at a breakneck pace, network activity types are diversifying and network behavior characteristics are getting more complicated, posing significant hurdles to intrusion detection [1], [2]. The ability to recognize distinct hostile network traffics, particularly unanticipated malicious network traffics, is a critical issue that cannot be overlooked. Avoided.

In fact, network traffic can be classified into two types (normal traffics and malicious traffics). Normal, DoS

(Denial of Service) attacks, R2L (Root to Local attacks), U2R (User to Root attack), and Probe are the five types of network traffic (Probing attacks). As a result, intrusion detection might be viewed as a categorization issue. Intrusion detection accuracy can be greatly enhanced by enhancing the efficiency of classifiers in efficiently identifying malicious traffics. In order to identify malicious traffic, machine learning approaches [3]– [8] have been widely employed in intrusion detection. These methods, on the other hand, are associated with shallow learning and frequently focus feature engineering and selection. They struggle with feature selection and are unable to properly tackle the huge intrusion data classification problem, resulting in low identification rates. accuracy and a high rate of false alarms In past years, various deep learning-based intrusion prevention systems have been proposed. [9] proposes a malware communication categorization approach based on a fully convolutional with traffic data represented as a picture. This approach does not require any manual design features and uses the original information as the classifier's input data. The authors of [10] examine the viability of Recurrent Neural Networks (RNN) for detecting network traffic behavior by modelling it as a series of changing states over time. The performance of the Long Short-term Memory (LSTM) network in identifying intrusion traffic data is verified by the authors in [11]. The results of the experiments reveal that LSTM can learn all of the approach classes. In the training data, there's a secret. All of the methods above consider network traffic as a whole, which is made up of a series of traffic bytes. They don't make full advantage of network traffic domain knowledge. CNN, for example, turns continuous network traffic into images for processing, which is the same as considering traffics as separate entities and ignoring network traffic internal relationships. To begin with, network traffic is organized in a hierarchical manner. Network traffic is a type of traffic made up of many data packets. A data packet is a type of traffic unit that is made up of several bytes. Second, traffic characteristics in the same and distinct packets differ dramatically. Separately extracted sequential features between distinct packets are required. To put it another way, not all traffic features are created equal. In the process of extracting features from a specific network communication, traffic classification is used.

1.2 Objective

Because normal activities predominate in real internet, the majority of traffic information is normal activity, with only a few destructive cyber-attacks, resulting in a large imbalance of types. The network is severely unbalanced and duplicated. Because everyday tasks predominate in real cyberspace, the vast majority of traffic data is ordinary traffic, with only a few destructive computer securities, resulting in a significant disproportion of classes. In a network that is very unbalanced and redundant.

2. Literature Survey

Naive Bayes vs decision trees in intrusion detection systems

Bayes systems are effective decision-making and reasoning aids when faced with uncertainty. Naïve Bayesian networks seem to be a very simple kind of Bayes networks that are extremely effective for prediction tasks. However, naive Bayes is dependent on a very stringent assumption. An experiment examination of the application of principal Component analysis (pact in malware detection is presented in this work. We show that, despite their modest structure, naïve Bayes can produce excellent results. KDD'99 invasion data sets are used in the experiment. According on whether we're dealing with full attacks, separating them into four broad categories, or merely focusing on ordinary and aberrant behavior, we evaluate three stages of attack granularity. Throughout the experiments, we evaluated the efficiency of naive Bayesian networks to that of well-known Bayes networks The decision tree is a well-known predictive model. Furthermore, we evaluate Bayes networks' incredible performance to the top results obtained on KDD'99.

Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS)

Due to its non - linearities and the numerical or qualitative networking traffic data stream with various aspects, the success of just about any Attack Detection (IDS) is a difficult task. Several sorts of intrusion detection methods have been developed to solve this problem, with varying degrees of accuracy. As a result, selecting an effective and reliable IDS method is a critical topic in data protection. We've created two models for identification in this project. The first is based on Classifiers (SVM), while the second is based on Random Forests (RF). The results of the experiments suggest that any classifier is effective. Such that the classifiers aren't biased more towards prominent impact in both the train and test sets Because the quantities of records in the train and test sets are now reasonable, it is

now feasible to perform the experiment on the entire collection rather than a tiny subset at random. The study findings will be extremely valuable in using SVM and RF in a more relevant approach in order to enhance performance while reducing false negatives.

Network Intrusion Detection Using Naive Bayes

Synopsis Network infrastructure is becoming more important than ever, thanks to the explosive rise of infrastructure activities and sensitive data on networks. In a network context, intrusion is a severe security issue. The ever-increasing number of novel infiltration types poses a significant difficulty for detection. The process of manually labelling the available network audit data instances is frequently arduous, time-consuming, and costly. In this research, we use nave bayes, one of the most effective data mining methods, to detect network intrusions based on anomalies. The originality of our approach in monitoring the network intrusion is demonstrated by experimental tests on the KDD cup'99 data set. When applied to KDD'99, the proposed technique performs better in terms of false positive rate, cost, and computing time When compared to a back propagation neural network-based technique, the data sets are far larger.

3. OVERVIEW OF THESYSTEM

3.1 Existing System

Machine learning has shown great performance in Feature Extraction (CV) [7] and Language Processing (NLP) [8] since Lacuna et al. [6] established the theory of Transfer Learning as a fundamental subfield of deep learning. Deep learning-based intrusion prevention system has been extensively researched in academia and industry. Deep learning refers to the process that uses training models to extract regional accents from subject to rapid and convert network data anomaly-based challenges into categorization challenges [9]. Learning approach of the difference from healthy and pathological behavior improves the actual improvement of intrusion treatment by training a huge number of data examples. However, in the number of co of network traffic, the classification discrepancy still has an impact.

3.1.1 Disadvantages of Existing System

The difficulty and focus of this grown to know is the safe and predicted reversion of a potential "degraded" condition to "Standard" before reaching the technical breakdown point F of the PF diagram.

• An engineering system, such as data, can show itself in two critical conditions in terms of safe operation: with or without problems." The authors define "failure" as an irregularity in the platform's functioning, or even just the platform's inability to accomplish the responsibilities for which it was created.

3.2 Proposed System

We do comprehensive data analysis cleaning using the original Proposed approach and the most recent CSECIC-IDS2018 as baseline methods. (2) This paper offers a machine learning technique for tackling the class imbalance problem in attack detection by decreasing the representativeness of the sample and enhancing the minority samples in the challenging set, allowing the classifiers to learn the distinctions better during training. (3) We divide the experiment into 30 techniques and use Random Forest (RF), Support Vector Machine (SVM), XGBoost, NLP, and other methods in the classification.

Advantages of Proposed System

✓ • This tool was used to determine the critical degree of network vectors in order to extract perfectly alright features that are more useful for detecting malicious traffic.

✓ • The characteristics created by the learning algorithm are then integrated into a fully connected for features extraction, resulting in the essential features that accurately represent network activity behaviors..

3.3 Proposed System Design

In this project work, I used five modules and each module has own functions, such as:

1. Data Collection
2. Preprocessing
3. Train Test Model Fit
4. Model Evaluate

3.3.1 Data Collection

Protocol type, flag, and application are the three metaphorical data kinds in NSL-KDD data features.

We map these attributes into binary vectors using a one-hot encoder. Processing in a single step: To convert symbolic features into numerical features, the NSL-KDD dataset is analyzed using a one-hot technique. The protocol type, for example, is the fourth attribute of the NSL-KDD data sample. Tcp, udp, and icmp are the three connection types. The one-hot approach is converted into a binary code that a computer can understand, with tcp being [1, 0, 0], udp being [0, 1, 0], and icmp being [0, 0, 1].

3.3.2 *Preprocessing*

Due to removal or input problems, a portion of the information comprises some noisy data, double values, incomplete data, infinite values, and so on. As a result, we begin by preparing the data. The following is the primary work. (1) Parallel values: remove the duplicated value from the sample and preserve just one valid data set. (2) Outliers: the sample group of null values (Not a Number, NaN) and unbounded values (Inf) in the sample data is minimal, therefore we remove them. (3) Delete and transform options: "Timestamp", "Form Of numerous", "Source Location", "Source Port", and other features are removed in CSE-CIC-IDS2018. We add two check dimensions if features "Init Bwd Win Byts" and "Init Fwd Win Byts" both have a value of 1 The value of one is one. Otherwise, the value is 0. To achieve this transformation in NSL-KDD, we are using the One Hot encoder. "TCP,'"UDP," and "ICMP," for example, are functionalities of three different protocol kinds. They generate binary vectors (1, 0, 0), (0, 1, 0) after One Hot encoding (0, 0, 1). The special product function is divided into 3 categories, with 11 flag functions and 70 service functions. As a result, the initial image vector with 41 dimensions now has 122 variables. (4) Numerical equivalence: The data is standardized, that is, the method of obtaining Z-Score, to alleviate the directional influence among both indicators and speed it up the back propagation algorithm and model convergence. As a result, this same average value of each feature will become 0 and also the input vector and model consolidation are accelerated When the deviation is changed to a normal random variable, it becomes 1, which is related to the entire sample allocation, and each piece of data might affect standardization. u seems to be the average of each feature, s seems to be the point difference of each function, and x 0 I is the element correlating to each column's features in the standardization formula...

3.3.3 *Train Test Model Fit*

Our dataset is now divided into training and testing sets. Our goal with doing this split is to evaluate our effectiveness of the algorithm on unknown data and to examine how well it has specialized on learning algorithm. Following that, a model fitting is performed, which is an important phase in the model development.

3.3.4 *Model Evaluate*

This is the final stage, in which we analyze how well your method fits on test data using various scoring metrics. I evaluated my prediction using the 'overall accuracy.' We start by creating a model instance, then use the knowledge for decision making to fit the trained information to the system, and finally use the predict method for making projections on the x test or testing data, which will be saved in a destination register y test hat. We'll input the y test and y test hat into the accuracy score procedure for model assessment and save the results in a variable called test accuracy, which will represent our model's accuracy score. These stages were taken for a variety of classification approaches, and the results were as follows related reliability test scores:

4 Architecture

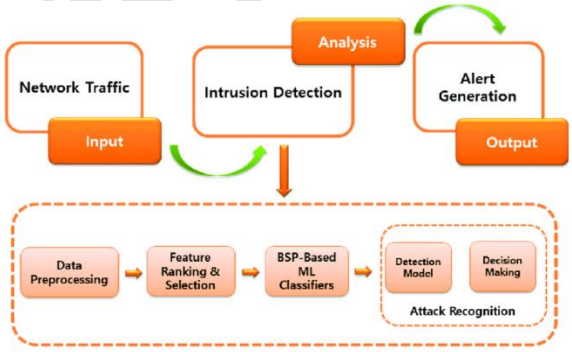
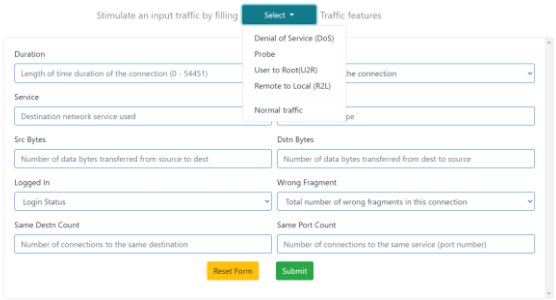


Fig 1: Frame work of cyber attack

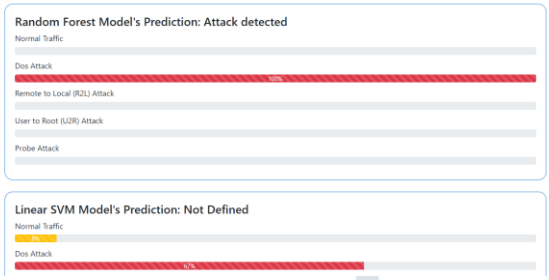
Above architecture describes process of network traffic analysis and data processing with feature extraction, model training, multiple algorithm accuracy comparisons. Network traffic is taken as input and intrusion detection of data is predicted by data processing through machine learning process.

5 RESULTS SCREEN SHOTS

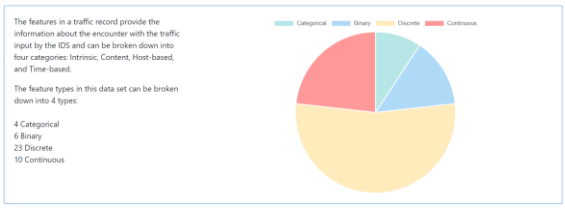
Attack Prediction Page:



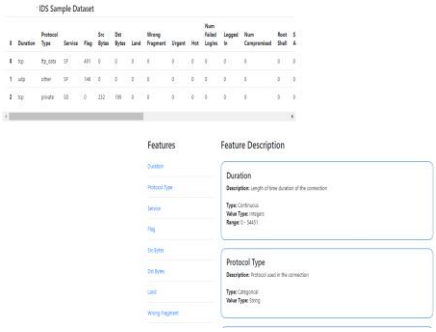
Predicted result:



Graphical Representation:



Data Features:



Attack Type Analysis:

7. CONCLUSION

✓ The burden on intrusion detection system is mounting as penetration test keeps on developing. Problems created by unbalanced internet traffic, in instance, make it difficult on security measures to determine the dispersion of targeted hackers, posing a significant danger to cyberspace security. This research introduced a new Difficult Set Samples Technique that allows the categorization model to learn from asymmetrical network data more effectively. A planned growth in the population of minority samples that must be taught can by above accuracy by reducing network traffic imbalance and strengthening minority knowledge under demanding samples. In machine learning algorithms, we integrated six classic classification techniques with various sample strategies. Experiments reveal that our method is capable of reliably predicting outcomes Evaluate which samples in the unbalanced internet traffic need to be enlarged in order to improve attack detection. After sampling the imbalanced training set samples using the MLP method, we observed that reinforcement learning outperformed ml algorithms in the research. Despite artificial neural network improve data interpretation, actual public datasets usually extract data characteristics in beforehand, making massive learning's ability to learn heavily processed components and take use of extracting features more constrained.

Future Enhancement

✓ In future network packet tracking tools can be used to track real time packets from any simulator and store data in excel sheet. This data can be used in training model and predict result for specific network traffic scenario.

8. References

- D. E. Denning, ‘‘An intrusion-detection model,’’ IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222–232, Feb. 1987.
- [2] N. B. Amor, S. Benferhat, and Z. Elouedi, ‘‘Naive Bayes vs decision trees in intrusion detection systems,’’ in Proc. ACM Symp. Appl. Comput. (SAC), 2004, pp. 420–424.
- [3] M. Panda and M. R. Patra, ‘‘Network intrusion detection using Naive Bayes,’’ Int. J. Comput. Sci.

Netw. Secur., vol. 7, no. 12, pp. 258–263, 2007.

- [4] M. A. M. Hasan, M. Nasser, B. Pal, and S. Ahmad, ‘‘Support vector machine and random forest modeling for intrusion detection system (IDS),’’ J. Intell. Learn. Syst. Appl., vol. 6, no. 1, pp. 45–52, 2014.
- [5] N. Japkowicz, ‘‘The class imbalance problem: Significance and strategies,’’ in Proc. Int. Conf. Artif. Intell., vol. 56, 2000, pp. 111–117.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, ‘‘Deep learning,’’ Nature, vol. 521, no. 7553, pp. 436–444, 2015.
- [7] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, ‘‘Deep learning for visual understanding: A review,’’ Neurocomputing, vol. 187, pp. 27–48, Apr. 2016.
- [8] T. Young, D. Hazarika, S. Poria, and E. Cambria, ‘‘Recent trends in deep learning based natural language processing [review article],’’ IEEE Comput. Intell. Mag., vol. 13, no. 3, pp. 55–75, Aug. 2018.
- [9] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, ‘‘A deep learning approach to network intrusion detection,’’ IEEE Trans. Emerg. Topics Comput. Intell., vol. 2, no. 1, pp. 41–50, Feb. 2018.
- [10] D. A. Cieslak, N. V. Chawla, and A. Striegel, ‘‘Combating imbalance in network intrusion datasets,’’ in Proc. IEEE Int. Conf. Granular Comput., May 2006, pp. 732–737..