DDOS DETECTION USING MACHINE LEARNING

TEAM MEMBERS

Akshat Garg <ag2193@rit.edu>

Bharadwaj Sharma Kasturi <bk5953@rit.edu>

Megha Gupta <mg9428@rit.edu>

Shaista Syeda <ss7810@rit.edu>

UNDER THE GUIDANCE OF

Prof. Nidhi Rastogi

Rigved, TA

Index

Contents

Introduction

Advances in technology have led millions of people to connect in some form of network and exchange critical data. Therefore, the need for security to protect the integrity and confidentiality of data is rapidly increasing. Efforts have been made to protect data transmissions, but attack technologies to infiltrate networks have continued to be developed simultaneously. Therefore, there is a need for a system that can adapt to these ever-changing attack techniques. In this paper, we have developed a system based on machine learning. Our goal is to find a suitable machine-learning algorithm to predict network attacks with the highest accuracy and develop a system to detect network intrusions using this algorithm. The algorithms compared are Naive Bayes, Decision Tables, K-nearest Neighbours, Random Forest, and AdaBoost. The dataset used to train the model is the KDD99 dataset. The reason I used machine learning is to give the system flexibility. For example, if a new type of attack is developed in the future, you can train your system to predict that attack. There are several types of intrusion detection systems, but our system is a knowledge-based intrusion detection system, also known as an anomaly-based system. Register anomalies and predict that such malicious networks will send alerts in the future. In this way, the network can be disconnected from such connections, and only secure connections are possible.

Problem Description

Information technology has advanced at a breakneck pace in the last two decades. Industry, business, and different aspects of human life all use computer networks. As a result, IT managers must focus on establishing reliable networks. On the other hand, the rapid advancement of information technology has created various problems in the laborious process of constructing trustworthy networks. Computer networks are vulnerable to various threats that jeopardize their availability, integrity, and confidentiality. One of the most widespread destructive attacks is the Denial-of-Service attack.

Data Research

This is the data set used for The Third International Knowledge Discovery and Data Mining Tools Competition.The task was to build a predictive model capable of distinguishing between ``bad'' connections, called intrusions or attacks, and “good” normal connections. The raw training data was about four gigabytes of compressed binary TCP dump data from seven weeks of network traffic. This was processed into about five million connection records. Similarly, the two weeks of test data yielded around two million connection records.

It is important to note that the test data is not from the same probability distribution as the training data, and it includes specific attack types not in the training data. This makes the task more realistic. Some intrusion experts believe that most novel attacks are variants of known attacks and the “signature” of known attacks can be sufficient to catch novel variants. The datasets contain a total of 24 training attack types, with an additional 14 types in the test data only.There was a data quality issue with the labels of the test data, also there was high imbalance in the data. Finally we learnt a classification model capable of distinguishing between legitimate and illegitimate connections in a computer network.

Literature Review

Being a classification problem, we came up with a supervised learning algorithm. Firstly, the class imbalance problem was overcome using a combination of oversampling as well as undersampling. When the skewness of the data was taken care of, various machine learning models were fed with the data to yield an efficient and usable yield. Algorithms like Regression, Naive Bayes, Decision Trees, Random Forest, Isolation Tree, XGBoost were utilized and conclusions were drawn considering the test scores like recall, precision and accuracy.

Analysis strategy

After studying the distribution of data, we identified that there were different subcategories of DDoS attack based on the layer of the network connection they attempt to attack. The result column had 60% of the normal data. and the rest 40% attack types were unevenly distributed. So, clearly the major challenge was to handle the class imbalance problem. Our approach was to implement different sampling techniques to get the classes balanced. Since, for class imbalance problems, accuracy is not an appropriate metric for model evaluation because the accuracy score would be high and heavily biased towards the majority classes (normal class for KDDCup dataset). Hence our main goal was to identify the minority class (attack sub-classes). So, we focused on precision-recall and FPR(fallout rate) for the evaluation of the machine learning models. In other words, our aim was to minimize a bad connection that gets classified as normal.

Analysis code

1. Data Exploration

Since our objective was to cover the majority of the attack types we combined the test and training data from the KDDCup dataset.

1. Identified the dataset for the null values. We found that there were no null values.

2. Checked for the duplicates in the data frame, around 70% of the data was duplicate so we dropped this.

3. Analyzed the attributes of the dataset and worked upon the numerical and categorical features individually.

2. Feature Selection

After plotting the correlation matrix, there were a total 9 pairs of highly correlated features, we selected one from each pair. After which there were a total 32 numerical attributes.

3. Data Preprocessing

1. Numerical attributes: total count=32

We standardized the numerical attributes which had the range greater than 1.

2. Categorical attributes: total count=3 (service, flag, protocol type)

For the columns service and flag had a high number of subcategories. On converting numerical value using one hot encoding result would have resulted in the addition of a column per subcategory. In this case it would result in adding 66 + 11 + 3 - 3 = 77 columns. This would have added to the complexity of the model. Hence, we used baseN encoding which highly reduces the dimensionality as the value of N increases.

3. We used SMOTE (Synthetic Minority Oversampling Technique) for balancing the classes.

4. Model Selection

We used the following algorithms for training the model and hyper tuned them.

1. Decision tree
2. Naïve Bayes
3. Random forest
4. Logistic regression
5. XGBoost

5. Model Comparison

Hypertuned Random Forest performed the best. (This needs to be completed will finalize this section after the python notebook)

Work Planning and organization of each team member

We devised this project as an opportunity not only to apply the techniques we have learned in the class, but also to broaden our knowledge on each component. Every member of the group contributed individually to this project, and whatever method was most effective according to them was used either in Data preprocessing, Data Cleaning or in Feature Selection.Everyone not only mastered the techniques they worked on, but taught them to others as well.

We collectively chose the best option to improve our model once everyone had finished their parts.

Improving teamwork and collaboration

From the initial steps of the project we as a team came up with our own inputs and had it discussed with the teammates in the weekly meetings. The most optimal methodology among the proposed ideas was considered and gave us the exposure of how a specific task could be tackled in different ways. This collaborative approach has helped us to learn collectively and help us have end to end knowledge of the project.

Individual Contribution

This project was a collective effort of me and my team members. We aimed at applying what we learnt from the lectures and got further reference from online resources. Predicting the DDOS attacks, a classification problem which we have proceeded using numerous supervised learning models. Starting from the collection of data, it was quite a challenge as there were misplaced labels and also the class imbalance problem(there was over 80% of the “attack” type and the rest was “normal” type). With the guidance of Rigved, I have come up with the conclusion to tackle these problems by renaming the labels and adding attack type entries which add on the weightage of the “normal” type in the train and test data. Later, when my teammates tried to come up with features for feeding the model. We’ve come up with a collective decision of doing the feature selection by building a correlation matrix and extracting the required features. Next comes the part which had a major contribution of mine, selecting the machine learning algorithms. Being a classification problem, the initial trial was to use the traditional Regression Algorithm. The data being imbalanced, accuracy couldn’t be relied on as a metric to consider. So, I went ahead and considered accuracy, precision and f1-score. After looking at the test scores, it was clear that Regression wasn’t the optimal algorithm to be used for the UDD Cup data. Then I tried the Naive Bayes algorithm which turned out to be similar to the Regression model in terms of the cost and performance. Then trees came into picture, Decision tree, Isolation tree and Random Forest. Decision tree was a boost to us, it turned out to be really good performing and was on the verge of sticking to it. But the high bias of the and the error rate of the tree pulled down the decision tree from best fit. As the Insolation tree is the best known algorithm for abnormality detection, I tried it reading from a research publication. But it didn’t turn out to be the way it was supposed to and had a poor score on the board. Finally, I went on with the Random forest model and was hoping it to be the best fit model for the dataset and as predicted it had the highest scores of all the machine learning models. Random forest gave out the best yield because the data used to train the model where we had to work data with a large number of features. With this model, it is faster to train compared to decision trees because we are working only on a subset of the features on the model, henceforth it supports the data with numerous features. Also another important reason that made random forest stand out was the way its default hyperparameters already return great results and the overfitting being avoided on the other hand. Also, I had an intuition to implement deep neural networks on this dataset, using the adams optimizer and a sequential model on the dataset. The multi modular neural network was overkill for a classification problem and helped me draw the conclusion that Random Forest was the most optimal model for the UDD Cup dataset. It was a great experience working on a project by myself from scratch and exploring this side of data science for the first time. Lastly, I would like to thank Prof. Nidhi Rastogi and the TA Rigved for their continuous guidance and valuable feedback which helped us reach the desired outputs in the given time.

THANK YOU