Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/5/20230686

Methods for DDoS attacks classification

Yiting Gao[1, 4, †], Jiayi Li[2,†], Xianghui Meng[3,†]

[1] Tsinghua University High school International, Beijing, 100101, China
[2] Guanggu School of No.1 Middle School Affiliated to CCNU, Wuhan, 430000, China
[3] Grace academy, Cearfoss Pike, Hagerstown, 21740, USA

[4] tinagao@thsi.edu.cn
[†] These authors contributed equally to this work

Abstract. At present, many people and companies are attacked by DDoS every day. DDoS attacks will make many large websites inoperable, which will not only affect people's normal use but also cause substantial economic losses. So far, many researchers have found different methods to prevent DDoS attacks, but there is no unified research on the classification of different types of DDoS attacks. If people are familiar with all common types of DDoS attacks and can successfully classify various DDoS attacks, they can take corresponding and effective precautionary measures to reduce risks and offer more protection for their own security. Therefore, the research topic is set to use four different algorithms to identify various DDoS attacks. The research methods of this paper are as follows: First, the CICDDOS2019 dataset is selected as the most basic research foundation. Second, the database is analyzed with K - nearest neighbors, Decision tree, Logistics Regression, and Naïve Bayes, then each method's accuracy and precision value are calculated. According to the results, the times that the highest accuracy and precision values of logistic expression are the most, and the times that the lowest accuracy and precision values of the decision tree are the most. Therefore, relative to K − nearest neighbors, Decision tree, and Naïve Bayes, the most effective way for people and companies is to classify the DDoS attacks by using Logistic Regression.

Keywords: DDoS classification, machine learning, K-nearest neighbors, decision tree, logistic regression.

1. Introduction

DDoS attacks have made a lot of people's lives harder. For example, the website is taking too long to load, the CPU usage is too high, and data transmission is interrupted [1]. In a lot of time, it looks like nonmalicious event and is hard to identify. People usually need a lot of traffic analysis to identify DDoS attacks. However, most of the papers show an abstract way to identify these attacks. Many methods can only let people know whether they have experienced DDoS attacks in a blurry way, but cannot give a professional way to classify DDoS attacks people have suffered and take corresponding solutions to recover losses. So our research is carried out to identify types of DDoS effectively and facilitate people to take correct precautionary measures and rescue measures. Many papers connected DDoS classification with machine learning, and they used methods like the random forest, AdaBoost and Naive Bayes. Most of them show very good accuracy in detecting and classifying different DDoS attacks. In

DOI: 10.54254/2755-2721/5/20230686

many resources researchers have found, they all have accuracy above 95%. Among different classification algorithms, KNN, logistic regression, decision tree, and Naive Bayes are the algorithms more familiar to different people compared to other algorithms. So, the study chose to focus only on these four analysis methods for research, described the principle of each method, listed the formulas used in each method, and gave the pseudo-code, logic of the process, and result tables of each algorithm's precision and accuracy.

2. Method

2.1. Data set

The data come from the CIC-DDOS2019 data set. It is collected from a well-known institution that has been focusing on cybersecurity: the Canadian Institute of Cybersecurity. There are a total of 50063112 records in the CIC-DDOS2019 data set. It includes 50006249 DDoS attacks and 56863 normal samples and has eleven different kinds of DDoS attacks (DrDoS_MSSQL, DrDoS_SYN, DrDoS_NetBIOS, DrDoS_DNS, DrDoS_SSDP, DrDoS_SNMP, DrDoS_LDAP, DrDoS_UDPLAG, DrDoS_TFTP, DrDoS_NTP, DrDoS_UD) and each group of these data contains 88 features. The data is trained and studied by three machine learning algorithms using the programming language Python. The data will be entered in the parameter of the function. Then the precision and accuracy will be calculated.

2.2. Algorithms

In this study, machine learning classification techniques are employed. The mechanisms by which a data set is transformed into a model are known as machine learning algorithms [2]. In supervised learning, people supply responses to a training data set, such as a collection of character photos and their names. A model that could correctly recognize a character would be the end result of the training. In order to identify DDoS assaults and provide categorical value predictions, K-nearest neighbors, decision trees, and logistic regression will be employed.

2.2.1. K-Nearest neighbors. A supervised machine learning algorithm is the k-nearest neighbors (KNN) [3 – 4]. It is a similarity-based classifier that makes the assumption that every pair of related data points belongs to the same group. Xk and Yk are the kth features in x and y, respectively. The number n denotes the total number of features. Between examples x and y, there is a standard Euclidean distance formula:

$$d(x,y)=\sqrt{\sum_{k=1}^{n}(x_k - y_k)^2} \tag{1}$$

2.2.2. Decision tree. Specifically, the smaller the probability p, the greater the final entropy (that is, the greater the amount of information) [5]. If the probability of an event in the extreme case is one, its entropy will become zero. For example, if people can predict the winning number of a lottery ticket, it will be developed. However, if one can predict that the sun will rise in the east tomorrow, it is worthless. In this way, the value of information can be measured by entropy.

$$H(x) = -\sum_{k=1}^{n} p(x_k)\log p(x_k) \tag{2}$$

2.2.3. Logistic regression. A classification approach for predicting binary classes is logistic regression [6-8]. The target variable's value is categorical. The algorithm makes predictions about the likelihood of binary classes using a logistic function. The sigmoid function is another name for the logistic function. The core of logical regression (LR) is to create a cost function in the face of a regression or classification issue, then iteratively solve the best model parameters using optimization techniques to test and validate the accuracy of the solved model.

$$S(z) = \frac{1}{1+e^{-z}}$$

$$y = \beta + \beta_1 X_1 + \beta_2 X_2 + \cdots\cdots + \beta_n X_n \tag{3}$$

DOI: 10.54254/2755-2721/5/20230686

2.2.4. Naive Bayes. A supervised machine learning approach for classification that is based on the Bayes theorem is called Naive Bayes [9]. The association between the independent feature vectors X1, X2,..., and Xn and the dependent class variable y is established via the Bayes theorem. P1 (x,y) denotes the likelihood that the data point (x,y) falls under category 1, and P2 (x,y) denotes the likelihood that the data point (x,y) falls under category 2. The following guidelines can then be applied to categorize a fresh data point (x,y):

If P1 (x,y) > P2 (x,y), the category is 1
If P2(x,y) > P1 (x,y), the category is 2

3. Result

The efficient indicators of the three machine learning classification algorithms (KNN, Decision Tree, Logistics Regression (LR) and Naïve Bayes) are evaluated with accuracy, precision [10].

1. TP (True Positive) real example: shows that both the fact and the prediction are accurate.
2. FP (False Positive) false positive example: shows that the prediction is accurate but the reality is not.
3. TN (True Negative): demonstrates that both the actual and the prediction are wrong.
4. FN (False Negative) false negative example: demonstrates that the reality is true and the prediction is wrong.

3.1. Accuracy

That is, the ratio that all predictions are correct:

$$ACC= frac \{TP+TN\} \{TP+FP+FN+TN\} \quad (4)$$

3.2. Precision

The precision ratio, the proportion of correct prediction as positive in all predictions as positive, and all predictions as the ratio of actual label 1 in 1:

$$Precision= \{TP\}/\{TP+FP\} \quad (5)$$

Table 1. DrDoS_MSSQL attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.998	0.999	0.999	0.999
Accuracy (percentage)	99.98	99.84	99.98	99.98

Table 1 reveals the evaluation indicators of classification algorithms on DrDoS_MSSQL dataset. Decision Tree, Logistic Regression (LR) and Naïve Bayes shows same precision values. KNN shows poorer precision value than Decision Tree, Logistic Regression (LR) and Naïve Bayes. Decision Tree gives us the lowest accuracy in this table and other three algorithms shows us same accuracy values.

Table 2. DrDoS_SYN attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.999	0.998	0.998	0.998
Accuracy (percentage)	99.98	99.97	99.98	99.98

Table 2 reveals the evaluation indicators of classification algorithms on DrDoS_SYN dataset. KNN shows the highest precision value in this table. Decision Tree, Logistics Regression (LR) and Naïve Bayes give us same precision values. Decision Tree shows us the lowest accuracy in this table and other three algorithms show the same accuracy values.

Table 3. DrDoS_NetBIOS attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.998	0.998	0.998	0.999
Accuracy (percentage)	99.95	99.91	99.95	99.95

Table 3 reveals the evaluation indicators of classification algorithms on DrDoS_NetBIOS dataset. Naive Bayes gives us the highest precision value in this table. KNN, Decision Tree and Logistics

DOI: 10.54254/2755-2721/5/20230686

Regression (LR) show us same precision values. KNN, Logistics Regression (LR) and Naïve Bayes has same accuracy values. The accuracy of Decision Tree is lower than KNN, Logistics Regression (LR) and Naïve Bayes.

Table 4. DrDoS_DNS attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.996	0.994	0.998	0.998
Accuracy (percentage)	99.88	98.38	99.89	99.89

Table 4 reveals the evaluation indicators of classification algorithms on DrDoS_DNS dataset. Decision Tree shows us the lowest precision value in this table. Logistics Regression (LR) and Naive Bayes give us same precision values. The accuracy value of Decision Tree gives us lower accuracy than other three algorithms. Logistics Regression (LR) and Naïve Bayes show the same accuracy values.

Table 5. DrDoS_SSDP attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.998	0.999	0.999	0.999
Accuracy (percentage)	99.95	99.92	99.94	99.94

Table 5 reveals the evaluation indicators of classification algorithms on DrDoS_SSDP dataset. KNN shows us poorer precision value than other three algorithms. Decision Tree, Logistics Regression and Naïve Bayes has same precision values. The accuracy of KNN is the best in this table.Logistics Regression(LR) and Naïve Bayes has same accuracy values.

Table 6. DrDoS_SNMP attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.999	0.999	0.999	0.999
Accuracy (percentage)	99.96	99.79	99.94	99.94

Table 6 reveals the evaluation indicators of classification algorithms on DrDoS_SNMP dataset. All the algorithms shows us same precision values. Logistics Regression (LR) and Naïve Bayes give us same accuracy values. Decision Tree shows us the lowest accuracy values than KNN, Logistics Regression (LR) and Naïve Bayes. The accuracy values of Logistics Regression (LR) and Naïve Bayes are same.

Table 7. DrDoS_LDAP attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.994	0.996	0.997	0.997
Accuracy (percentage)	99.90	99.60	99.91	99.91

Table 7 reveals the evaluation indicators of classification algorithms on DrDoS_LDAP dataset. KNN shows us the lowest precision value than Decision Tree, Logistics Regression (LR) and Naïve Bayes. Logistics Regression and Naïve Bayes give us same precision values. The accuracy values of Logistics Regression (LR) and Naïve Bayes are same. Decision Tree shows us lower accuracy value than KNN, Logistics Regression (LR) and Naïve Bayes.

Table 8. DrDoS_UDPLAG attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.990	0.989	0.991	0.991
Accuracy (percentage)	99.62	99.44	99.62	99.62

Table 8 reveals the evaluation indicators of classification algorithms on DrDoS_UDPLAG dataset. Decision Tree shows us poorer precision value than KNN, Logistics Regression (LR) and Naïve Bayes. Logistics Regression (LR) and Naïve Bayes shows us same precision values. Decision Tree also shows us lower accuracy than KNN, Logistics Regression (LR) and Naïve Bayes. KNN, Logistics Regression (LR) and Naïve Bayes shows us same accuracy values.

DOI: 10.54254/2755-2721/5/20230686

Table 9. DrDoS_TFTP attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.997	0.997	0.997	0.997
Accuracy (percentage)	99.95	99.82	99.96	99.96

Table 9 reveals the evaluation indicators of classification algorithms on DrDoS_TFTP dataset. KNN, Decision Tree, Logistics Regression (LR) and Naïve Bayes all shows us same precision values. The accuracy value of Decision Tree is lower than KNN, Logistics Regression (LR) and Naïve Bayes.Logistics Regression (LR) and Naïve Bayes also gives us same accuracy values.

Table 10. DrDoS_NTP attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.989	0.988	0.999	0.989
Accuracy (percentage)	99.65	98.80	99.66	99.65

Table 10 reveals the evaluation indicators of classification algorithms on DrDoS_NTP dataset. Decision Tree gives us poorer precision value than KNN, Logistics Regression (LR) and Naïve Bayes. Naïve Bayes and Logistics Regression (LR) also show us same precision values. The accuracy of Decision Tree is higher than KNN, Logistics Regression (LR) and Naïve Bayes. KNN and Naïve Bayes also show us same accuracy values.

Table 11. DrDoS_UDP attack detection results.

Classification algorithm	KNN	Decision Tree	LR	Naive Bayes
Precision	0.997	0.997	0.998	0.998
Accuracy (percentage)	99.94	99.80	99.93	99.93

Table 11 reveals the evaluation indicators of classification algorithms on DrDoS_UDP dataset. KNN and Decision Tree show us same precision values that are lower than Logistics Regression (LR) and Naïve Bayes. Logistics Regression (LR) and Naïve Bayes show us same precision values that are higher than KNN and Decision Tree. Decision Tree shows us poorer accuracy value than KNN, Logistics Regression (LR) and Naïve Bayes. Naïve Bayes and Logistics Regression (LR) shows us same accuracy values.

4. Conclusion

By comparing the results of the four detection methods in the table, readers can see that the accuracy and precision of the Decision tree are significantly lower than those of the other three algorithms, and the results of Logistics Regression and Naive Bayes are always highly consistent. So it is more effective for people to use KNN, Logistics Regression and Naive Bayes to identify the types of DDoS attack and take corresponding precautionary measures. Decision tree may not be a good choice compares to other algorithms. However, overall, most of the algorithms has a high accuracy and precision, confirming that machine learning can combine with cypersecurity and bring more efficiency when helping people.

Cyber attacks are becoming more and more important while technologies are growing so fast. Society should maximize the positive effects of machine learning. This study only focused on four machine learning algorithms and only evaluated their accuracy and precision. Future study could analyze on other machine learning algorithms, and give more information to validate the effectiveness of the algorithms, not just accuracy and precision. It should also be noted that there are limitations to the data presented in the study. The study used data set from years ago. There are still new attacks appearing during these years and the study didn't get a chance to analyze that. Next research should make sure the data set is up to date to make more accurate study.

Appendix

Classifier algorithm (test data, training data, target variable set, k):

using linear operation to calculate the distance between each record of the training data, and the test data

sorting the value of the distance

DOI: 10.54254/2755-2721/5/20230686

```
for i in  range(k):
calculate the number of categorical variables for each of the first k categorical variables
returns the variable with the most numbersClassifier algorithm (test data, training data, target variable set, k):
 using linear operation to calculate the distance between each record of the training data, and the test data
 sorting the value of the distance
for i in  range(k):
calculate the number of categorical variables for each of the first k categorical variables
returns the variable with the most numbers

# D = {(x1, y1) . (x2, y2) .
• (xm, yn)} is a dataset
#A II {al, a2, a3. is the attribute set for dividing nodes
Node # has two main attributes: content represents that the node needs classification
def generaterree (D,A):
NewNode=null # Generate a new node
# If the current dataset is of the same category, set it as a leaf node and
The data in I D belong to category C:
newNode.  content = D
newNode.  type = C
return
#If there are no attributes or the dataset is displayed in the remaining attributes
If A=empty set o Data in D have the same value in A:
newNode.  content
=D
newvode. The most classes in type two D
return
# Select the best severity from A
a=selectBestPorperty(A)
#Is every of a
-Each value generates a node, which is processed recursively
Each value of for a res[i]:
Generate a new branch node node[i]
D[i]=data with res [i] in D
node[i].content = D[i]
if node[i].content
== nu11:
node[i]. Type=Most classes in D
else:
generateTree (D[i],A - {a})
Return

int main()
{
Read the training set and the test set, where the first two are taken as the training set and the third as the validation set for every three samples.
Initialization w:for(int i=0; i<Length; i++) w[i]=1;
for(int k=0:7)
for(int i=0:traincnt)//traverse the training set sample
{
```

DOI: 10.54254/2755-2721/5/20230686

```
CalWeight(i); // Calculate the weight fraction of sample i
CalCost(i); // gradient (cost) calculation of each dimension
Updatew(); Update w
if(i%20==0) // accuracy is calculated once every w20 times updated
{
Predict();//Predict validation set sample
Cal_acc();// Calculate accuracy
ac[cnt]=accuracy;
cnt++;
}
}
output_result();//output the accuracy of the verification set for debugging
output_test_result();//Output test set predictions
}
void CalWeight(int index){
weight=transpose of the current vector w * sample i vector;
}
void CalCost(int index){
Calculate the gradient of each dimension, stored in the vector array Cost[];
}
void Updatew(){
Use w= w - alpha x gradient to update the regression coefficient(w)
}
void Predict(){
P=1/(1+exp(-1*w^T*sample i vector);
if(P>0.5) p_label=1;
else p_label=0;
```

Input: $n \geq 0 \vee x \neq 0$
Output: $y = x^n$
$y \leftarrow 1$
If $n < 0$ then
$X \leftarrow 1/x$
$N \leftarrow -n$
else
$X \leftarrow x$
$N \leftarrow n$
end if
while $n \neq 0$ do
if N is even then
$X \leftarrow X \times X$
$N \leftarrow N/2$
else {N is odd}
$y \leftarrow y \times X$
$N \leftarrow N - 1$
end if
end while

References

[1] “Symptoms of a Ddos attack on a server_chenghuangsi2633’s blog -CSDN blog.” Blog.csdn.net,

DOI: 10.54254/2755-2721/5/20230686

blog.csdn.net/chenghuangsi2633/article/details/101034236. Accessed 6 Oct. 2022.

[2] Heller, Martin. “Machine Learning Algorithms Explained.” InfoWorld, 9 May 2019, www.infoworld.com/article/3394399/machine-learning-algorithms-explained.html.

[3] S. Dong and M. Sarem, "DDoS Attack Detection Method Based on Improved KNN With the Degree of DDoS Attack in Software-Defined Networks," in IEEE Access, vol. 8, pp. 5039-5048, 2020, doi: 10.1109/ACCESS.2019.2963077.

[4] Dasari, K.B., Devarakonda, N. (2021). Detection of different DDoS attacks using machine learning classification algorithms. Ingénierie des Systèmes d’Information, Vol. 26, No. 5, pp. 461-468. https://https://doi.org/10.18280/isi.260505

[5] Sharafaldin I, Lashkari A H, Hakak S, et al. Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy[C]//2019 International Carnahan Conference on Security Technology (ICCST). IEEE, 2019: 1-8.

[6] Yan, D. Tang, S. Zhan, R. Dai, J. Chen and N. Zhu, "Low-Rate DoS Attack Detection Based on Improved Logistic Regression," 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2019, pp. 468-476, doi: 10.1109/HPCC/SmartCity/DSS.2019.00076.

[7] “How does a logistic regression algorithm work? Baidu knows.” Zhidao.baidu.com, zhidao.baidu.com/question/753167760030154412/answer/3658920465.html?fr=zywd.Accessed 24 Sept. 2022.

[8] “The principle of logistic regression is deduced in detail. BiKongZhiGe’s blog -CSDN blog_ principle of logistic regression” Blog . csdn.net, blog.csdn.net/qq_38923076/article/details/82925183. Accessed 24 Sept. 2022.

[9] “Machine learning ： Naive Bayes(Naive Beyes)_Panghuaichunhui’s blog -CSDN’blog.” Blog.csdn.net,blog.csdn.net/weixin_42479155/article/details/103329046?csdn_share_tail=%7B%22type%22%3A%22blog%22%2C%22rType%22%3A%22article%22%2C%22rId%22%3A%22103329046%22%2C%22source%22%3A%22unlogin%22%7D. Accessed 10 Oct. 2022.

[10] “Accuracy, precision, recall and F value in some algorithms” https://www.codenong.com/cs105954268/