Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 8, No. 2, 2023

A Network Intrusion Detection Method Based on a Fast KNN Algorithm

Pengcheng He, Guifeng Feng*, Huixu Li, Le Yang

Foshan Network Security Emergency Command Center, Foshan, 528000, China
* Corresponding author: Guifeng Feng (Email: 1215458536 @qq.com)

Abstract: This paper proposes a network intrusion detection method based on fast KNN algorithm. First, continuous data discretization processing and character data digitization processing are performed on the original data. Then the feature reduction algorithm based on mutual information is used to reduce the features of the preprocessed data. Finally, the fast KNN algorithm is used to classify and detect the feature-reduced data, and the classification results are output. The above method was verified on the KDD CUP99 data set and compared with existing technologies. The method proposed in this article clearly has the advantages of high classification efficiency and classification accuracy.

Keywords: Network security, Intrusion detection, K-neighbor algorithm, Machine learning.

1. Introduction

With the complexity, diversification and intelligence of various computer network attack means, the problem of network information security has become increasingly prominent. Benefit to network destruction of terminal operating system, illegal theft of personal data, bank account password, illegal intrusion of system database and other behaviors seriously hinder the normal use of the Internet. Intrusion detection technology as an important dynamic protection means of network security system, can identify the computer network illegal or malicious attacks, and the corresponding response, as a network security technology, and the second security gate after the firewall, intrusion detection technology is one of the very important core technology of Internet security, it extends the system administrator security management ability at the same time can improve the integrity of the system security structure. Intrusion detection algorithm is the most core part of the intrusion detection system. Its detection ability and efficiency directly determine the detection ability of the whole intrusion detection system.

The k-nearest neighbor algorithm, also called KNN or k-NN, is a non-parametric, supervised learning classifier where KNN uses proximity to classify or predict groups of individual data points. The existing KNN algorithm and related improved algorithms have been widely used in network intrusion detection[1-6]. However, there is still some room for improvement in both the detection capacity and the detection efficiency. It is of great significance to improve the classification accuracy, reduce the false detection rate, and maximize the learning speed of the algorithm. The fast KNN algorithm proposed in this paper is improved in terms of efficiency and accuracy, and is validated on the KDD 99 dataset.

2. The KNN Invasion Detection Model

2.1. The k-nearest neighbor algorithm

The k-neighbor algorithm (KNN algorithm) was proposed by Thamas in 1967. It is based on the following idea; to determine the category of a sample, you can calculate the distance from all training samples, then find the closest k samples, count the number of categories of the sample, the largest class is the classification result, for the classification problem, given a training sample (x_i, y_i), where x_i is the feature vector, y_i is the label value. Set parameter k, assume the type is c, and the feature vector of the sample to be classified is x. The flow of the prediction algorithm is as follows:

(1) Find the k samples closest to x in the training sample set, assuming that the set of these samples is N.

(2) Count the number of samples of each class in the set N. C_i, $i = 1,2, \ldots, e$.

(3) The final classification result is the arg maxCi.

Where $\arg\max C_i$ representing the largest value corresponding to the class i, if k=1, the k nearest neighbor algorithm degenerate into the nearest neighbor algorithm.

2.2. Intrusion detection model based on the fast KNN algorithm

The structure of the invasion detection model based on the fast KNN algorithm is shown in Figure 1, as follows:

Step 1: Data preprocessing step: receive the intrusion detection raw data, conduct data preprocessing of the raw data, including continuous data discretization processing and character data digital processing.

Step 2: Feature reduction step. The feature reduction algorithm based on mutual information is used to reduce the pretreated data.

Step 3: Use the fast KNN algorithm to classify the data after feature reduction and output the classification.

The feature reduction step of step 2 specifically includes the following sub-steps:

(1) Initialization: The feature set of the original data is set to $F(f_1, f_2 \cdots f_m)$, m is the total number of features. The category identifier of the data set is set to y. Set the empty set S, assuming that N features need to be selected.

(2) Select the first feature: for each feature f_i in F, calculate the mutual information $I(f_i;y)$ between f_i and the category identifier y, then choose f_i that maximizes the value of

$I(f_i;y)$, store f_i in the set S, this feature is the first feature, and remove f from the set F.

(3) The remaining N-1 features were selected in turn: Select the q-th feature using the "minimum redundancy-maximum correlation" standard strategy:

$$I_q = \arg\max_{1\le i\le m}\left\{ I(f_i;y) - \frac{1}{q-1}\sum_{f_i\in S_{q-1}} I(f_i;f_t) \mid f_i \in F \right\}$$

In the formula, term $I(f_i;y)$ is the "maximum correlation" condition, I_q represents the mutual information of the q features, and S_{q-1} represents the feature subset containing the selected $q-1$ features.

4) Output the selected feature subset *S*.

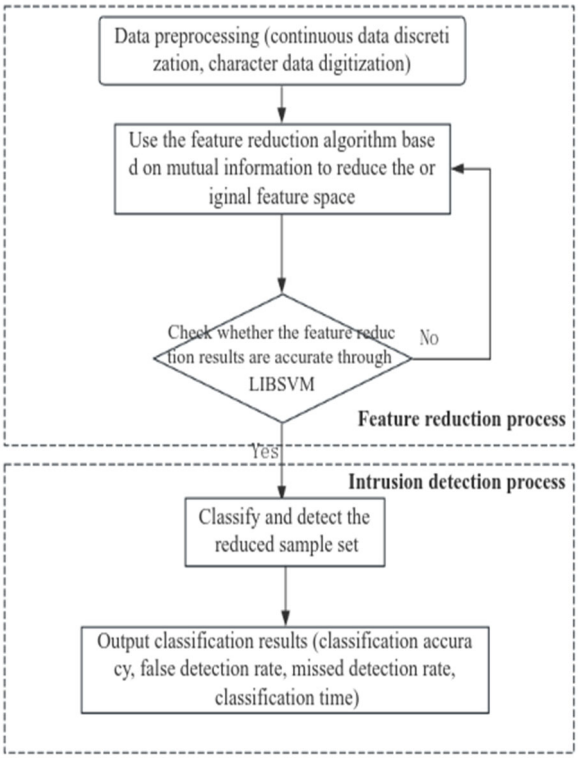

Figure 1. Structure of the intrusion detection model based on the fast KNN algorithm

The fast KNN algorithm is used to classify and detect the data after feature reduction:

(1) Obtain the training sample set, and delete the duplicate data in the training sample set;

(2) Establish the index model;

(3) For the current samples to be classified, judge whether there are samples with the same samples in the classified sample set as the samples to be classified; if so, the category identification of the same classified samples will be directly output; if no, perform Step 4);

(4) Quickly search for the k nearest neighbors of the samples to be classified in the training sample set according to the established index model;

(5) Output the category identification of the sample to be classified according to the k nearest neighbors.

3. Experiments

In the experimental section, the proposed method in this paper is compared with the conventional KNN algorithm. Experiments were performed based on the KDD 99 dataset. The evaluation criteria include classification accuracy, misdetection rate and missed detection rate.

As the original KDD CUP99 dataset is too large, only 80056 of them are randomly selected for the study, 40000 are used as training samples and 40056 were used as test samples. Then, the selected training sample data and test sample data are preprocessed in the same mode, including: discretization of continuous data and digitization of character data. The feature reduction algorithm based on mutual information is then used to reduce the collated sample set with a feature space of 41 dimensions. Then the training sample data set is repeated by reducing the data, the number of training samples is greatly reduced, and then the improved fast KNN classification algorithm (i. e., index model, cache function) is used for classification detection, and finally the required results are obtained.

Size reduction of the preprocessed KDD CUP99 dataset using a mutual information-based feature reduction method. In the experiment, no matter the 2 classification data set or the 5 classification data set, when the features are taken above 5 dimensions, the LIBSVM classification tool can be maintained above 98%. To speed up the classification when using KNN classification, we choose to reduce the feature dimension to 5 dimensions. And it was then used for subsequent invasion detection pattern classification studies.

3.1. Verify the algorithm learning speed and training sample size correlation

To verify the correlation of the learning speed of the KNN algorithm with the training sample library size. Now, based on the uncensored training sample library and the deleted training sample library as experimental data, the KNN algorithm is uniformly used for classification learning. In order to save the experimental time, only the k values are 4 and 10, and the effect of the deletion of the training sample library on the learning speed is studied. The experimental results are shown in Table 1.

Table 1. Comparison of experimental results before and after deletion of the training sample library

Algorithm	k	Classify	Training set	Accuracy	Mischeck rate	Loss	Time /s
Original KNN	4	2	Uncut	0.9621	0.1581	0.0113	3.1739e+004
			Deleted	0.9638	0.1452	0.0120	2.5936e+003
		5	Uncut	0.9565	0.1893	0.0112	3.216le+004
			Deleted	0.9504	0.2196	0.0120	2.5643e+003
	10	2	Uncut	0.9629	0.1536	0.0114	4.1131e+004
			Deleted	0.9632	0.1507	0.0117	3.2978e+003
		5	Uncut	0.9596	0.1705	0.0116	4.1032e+004
			Deleted	0.9591	0.1704	0.0123	3.3002e+003

It can be seen from Table 1, if the duplicate data in the training sample library is not censored, the time cost of the whole learning process is very huge, which is very undesirable. However, after deleting the duplicate data of the training sample library, the time spent by the algorithm learning is significantly shortened by about 13 times. This is because KNN is a pattern classification method based on distance calculation, so the larger the training sample library, the larger the calculation amount of KNN will be, and the longer the corresponding classification learning time will be. At the same time, it can be seen from the experimental data that before and after the deletion of the training sample library, the classification accuracy of the algorithm did not change greatly either for the 2 classification data set or the 5

classification data set.

3.2. Performance comparison of the fast KNN algorithm and the KNN algorithm

In order to verify the superiority of the fast KNN algorithm in speed, different nearest neighbor k values are now selected, and the KNN algorithm and the fast KNN algorithm are used for classification learning. In order to save the experimental time, the two classification learning algorithms use the training sample library after deleting the repeated data.

(1) 2 Classification status:

When the experimental data are 2 classification cases, which is normal and abnormal, the experimental results of KNN versus fast KNN are shown in Table 2.

Table 2. Comparison of the 2-classification experiment results between KNN and fast KNN

K	Algorithm	Classification accuracy	Mischeck rate	Loss	Time /s
10	KNN	0.9632	0.1507	0.0117	3297.8
	fast KNN	0.9774	0.0875	0.0082	206.9
8	KNN	0.9636	0.1481	0.0117	3044.0
	fast KNN	0.9903	0.0059	0.0105	194.6
6	KNN	0.9637	0,1467	0.0119	2804.4
	fast KNN	0.9907	0.0044	0.0104	183,7
4	KNN	0.9638	0.1452	0.0120	2593.6
	fast KNN	0.9908	0.0034	0.0105	170
2	KNN	0.9600	0.1349	0.0191	2500.2
	fast KNN	0.9903	0.0021	0.0113	137.8

The four performance indexes of KNN and fast KNN under 2 classification: classification accuracy, false detection rate, omission rate and running speed. It can be concluded that: when k=4, the fast KNN classification algorithm has a relatively high performance.

(2) 5 Classification status:

When the experimental data is 5 classification cases, that is, normal and 4 attack types, the experimental results of KNN and fast KNN are shown in Table 3.

Table 3. Comparison of the 5-classification experiment results between KNN and fast KNN

K	Algorithm	Classification accuracy	Mischeck rate	Loss	Time /s
10	KNN	0.9591	0.1704	0.0123	3300.2
	fast KNN	0.9685	0.1216	0.0115	147.4
8	KNN	0.9586	0.1736	0.0121	3056.7
	fast KNN	0.9721	0.1164	0.0083	131.1
6	KNN	0.9570	0.1831	0.0120	2822.6
	fast KNN	0.9738	0.1098	0.0077	110.5
4	KNN	0,9504	0.2196	0.0120	2564.3
	fast KNN	0.9633	0.1703	0.0071	94.3
2	KNN	0.9078	0.4273	0.0181	2344.6
	fast KNN	0.9550	0.2166	0.0070	75.8

Comprehensive analysis of four performance indicators of KNN and fast KNN under 5 categories: classification accuracy, false detection rate, missed detection rate and running speed. It can be concluded that when k=6, the fast KNN classification algorithm has relatively high performance.

From the experimental results in 3.1 and 3.2, it can be concluded that the improved fast KNN algorithm not only improves the classification accuracy, but also the classification speed is about 200 times faster than the traditional KNN algorithm.

4. Conclusion

There is a large amount of unlabeled network data in the Internet environment, and intrusion detection systems need to detect and classify various intrusion events. This paper proposes a fast KNN algorithm to address the shortcomings of the traditional KNN algorithm. This feature reduction algorithm based on mutual information reduces the high-dimensional feature set of the original data, removes redundant information and interference information in the feature set, and improves the performance of the KNN algorithm. Judging from the experimental results, the improvements of the improved algorithm are as follows: by deleting the training sample library and reducing the training sample set, the consumption of algorithm learning time is reduced to a large extent and the efficiency is accelerated. By establishing an index model and using caching technology, the search range and the number of disk starts are reduced, and the time to search for k nearest neighbors is reduced, thereby greatly speeding up classification, improving the efficiency of the KNN algorithm, and shortening the classification time.

References

[1] Guo, Gongde, et al. "KNN model-based approach in classification." On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings. Springer Berlin Heidelberg, 2003.

[2] Zhang, Shichao, et al. "Learning k for knn classification." ACM Transactions on Intelligent Systems and Technology (TIST) 8.3 (2017): 1-19.

[3] K. Mohammed, Abdul Hanan, et al. "Iot cyber-attack detection: A comparative analysis." International Conference on Data Science, E-learning and Information Systems 2021. 2021.

[4] Pathak, Ashwini, and Sakshi Pathak. "Study on decision tree and KNN algorithm for intrusion detection system." International Journal of Engineering Research & Technology 9.5 (2020): 376-381.

[5] Rak, Ewa, et al. "The distributivity law as a tool of k-NN classifiers' aggregation: mining a cyber-attack data set." 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). IEEE, 2020.

[6] Sakhnini, Jacob, Hadis Karimipour, and Ali Dehghantanha. "Smart grid cyber attacks detection using supervised learning and heuristic feature selection." 2019 IEEE 7th international conference on smart energy grid engineering (SEGE). IEEE, 2019.