Feature Selection and Deep Learning based Approach for Network Intrusion Detection

Jie Ling [a], Chengzhi Wu [b, *]

Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China.

[a] jling@gdut.edu.cn, [b, *] chengzhi_92@126.com

Abstract. The intrusion detection system deals with huge amount of data containing redundant and noisy features and the poor classifier algorithm causing the degradation of detection accuracy, in this paper, we introduce the random forest feature selection algorithm and propose a method that multi-classifier ensemble based on deep learning for intrusion detection. It used the random forest feature selection algorithm to extract optimal feature subset that are used to train by support vector machine, decision tree, naïve bayes and k-nearest neighbor classification algorithm, then, applying the deep learning to stack the output of four classifiers. The experimental results show that the proposed method can effectively improve the accuracy of intrusion detection compared with the majoring voting algorithm.

Keywords: Intrusion detection, Random forest, Deep learning, Feature selection.

1. Introduction

With the popularization of Internet technology in politics, economy and military, the network environment is increasingly complex, presenting complexity and diversifycation. It is difficult to meet the needs of network security by relying on traditional network defense technologies. Network Intrusion Detection System (NIDS), as an active defense technology, has become one of the research contents of many scholars. It mainly detects and analyzes data in the network stream, related log and audit files to determine whether the behavior violates security policies and computer system security [1].

Network intrusion detection can be considered as a classification problem to a certain extent, mainly to solve the problem of data dimensionality reduction, classifier construction and the model optimization to improve the accuracy of intrusion detection. The goal of feature selection is to select the optimal feature subset, reduce the data dimension to improve the detection efficiency. Therefore, some scholars have proposed the fast feature selection algorithm based on graph clustering, particle Swarm Optimization algorithm and Genetic Algorithm [2-4]. The machine learning has become the hottest topic of network intrusion detection [5], mainly including Bayesian network, Support Machine Learning (SVM), neural network [6-7]. In recent years, many scholars have cited intelligent optimization algorithms into the optimization of classify-cation algorithm. Because the predictive and generalizing ability of SVM algorithm mainly depends on kernel function parameters and penalty factor se-lection, optimization algorithms such as Binary Quantum Inspired Gravity Search Algorithm, Chaotic Universal Gravitation Search and Ant Colony Algorithm [8-10] have been used to optimize the SVM to improve the accuracy of classification. In the field of artificial intelligence, neural network algorithm is also a research hotspot in recent years. Based on Genetic Algorithm and Artificial Bee Colony [11-12] optimization algorithms have been used to optimize the weight and threshold of neural network to improve the accuracy of detection. Therefore, to improve the accuracy of intrusion detection, it is vital to have an optimal feature subset and a good classification algorithm.

Based on the above analysis, this paper introduces the feature selection algorithm of random forest to reduce the dimension of data, remove redundant features, obtain the optimal feature subset, and propose a deep learning ensemble method for the multi-classifiers, which is used for classification of intrusion detection.

2. Methods

2.1 Feature Selection

The network data always contains many features that are irrelevant or contain little information. These features have little effect on the classification results. The feature selection algorithm mainly eliminates these redundant features and reduces the impact of redundant features on the classification algorithm.

2.1.1 Correlation Feature Selection Algorithm (CFS)

CFS [13] algorithm mainly searches for feature subsets based on the redundancy between features. The purpose is to find feature subsets with high correlation and low correlation between features. The algorithm overcomes the single variable screening and fully considers the interdependence of features can effectively eliminate features that are not related. The feature subset evaluation function of CFS is as follows:

$$Merit_s = \frac{k r_{\bar{c}f}}{\sqrt{k+(k-1) r_{\bar{f}f}}} \tag{1}$$

Where Merit_s is a heuristic ‘merit’ containing a feature subset S of k features, $r_{\bar{c}f}$ is a feature-class average correlation, and $r_{\bar{f}f}$ is a feature-feature average correlation. r is the Pearson correlation coefficient and all variables need to be normalized.

2.1.2 Random Forest Feature Selection Algorithm

Random Forest algorithm (Random Forest) is an integrated learning method. It uses random resampling technique bootstrap and node stochastic classification technology to construct multiple decision trees and obtain the final classification result by voting. RF has the ability to analyze complex classification features, is robust to redundant data and missing data, and is fast to learn. Variable importance measure is an important feature of random forest algorithm. It can be used as a tool for high-dimensional data feature selection. It usually provides four variables importance measure methods. In this paper we apply an Area Under the Curve (AUC) based permutation variable importance measure (VIM) for feature selection [14].

An AUC-based permutation VIM mainly indicates the importance of the variable by calculating the average reduction before and after the area under the curve after the slight disturbance of the data independent variable. Therefore, the variable importance measure $VI_{x_j}^{AUC}$ of x_j predictor is calculated as follows:

$$VI_{x_j}^{AUC} = \frac{1}{ntree^*} \sum_{t=1}^{ntree^*} (AUC_{tj} - AUC_{t\bar{j}}) \tag{2}$$

ntree^* indicates the trees in the forest, AUC_{tj} indicates the AUC calculated from the OOB observations before permuting predictor x_j in tree t, $AUC_{t\bar{j}}$ indicates the AUC calculated from the OOB observations after permuting predictor x_j randomly in tree t.

2.2 Classification Algorithm

(1) Naive Bayes classifier is one of the classic models in the classification field due to its simplicity and high computational performance. Based on Bayes' theorem, it is assumed that the influence of each feature parameter on a given type is independent of each other, and the classification result is identified by known prior probability and conditional probability. The advantage is that it is insensitive to missing data and can quickly and efficiently obtain classification results.

(2) Support Vector Machine (SVM) is based on the principle of structural risk minimization, looking for an optimal interval to divide the instance into two categories.

Suppose $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ is the training samples, so that its objective function:

$$\min \frac{1}{2} ||w|| + c \sum_{i=1}^{n} (\xi_i + \xi_i^*) \tag{3}$$

constraints:

$$\begin{gathered} y_i - (w\phi(x_i) + b) \leq \varepsilon - \xi_i \\ (w\phi(x_i) + b) - y_i \leq \varepsilon - \xi_i^* \\ \xi_i, \xi_i^* \geq 0, i = 1,2, \dots., n; c > 0 \end{gathered} \tag{4}$$

Where c is the penalty factor and ε is not sensitive to the loss function parameter. ξ_i and ξ_i^* are non-negative slack variables. So the final difference function is:

$$f(x) = \sum_{i=1}^{n} (a_i - a_i^*) K(x_i, x_j) + b \tag{5}$$

a_i and a_i^* are determined by a penalty factor $\mathrm{K}(x_i, x_j)$ is a kernel function that satisfies the Mercer condition, and b is a threshold.

(3) The decision tree classification algorithm follows the principle of “divide and conquer” and is an efficient classification algorithm. Its classification learning is mainly divided into two stages.

(a)Tree construction stage: adopts top-down recursive method, starting from the root node to test attributes according to the given selection at each node, then establishing branches according to the possible values of the corresponding attributes, and dividing the training set until one node All samples are divided into one class, or the number of samples of a node is less than a given value.

(b)Tree pruning stage: The pruning process attempts to eliminate noise or isolated points in the training data to improve the accuracy of classification of unknown data sets. Tree pruning mainly includes first pruning and post pruning. The standard has a minimum description length principle and a minimum expected error rate principle.

(4) KNN is a simple and effective method for target classification based on the most recent training samples in the feature space. When the prior knowledge about the data distribution is little or no prior knowledge, the KNN classifier transforms the samples into the metric space and classifies the new points based on the majority of the votes obtained from the K nearest points in the training data. Usually, the Euclidean distance is often used as a distance metric to measure the similarity between two vectors.

2.3 Proposed Method

In practice, classical methods always lead to over-fits or lower performance by algorithm shortcomings, so ensemble of multiple different classification algorithm may improve performance over individual algorithm. In this paper, we adopt the deep learning to stacking the multiple model.

The deep neural network algorithm is inspired by the brain and is widely used. It includes the input layer, the hidden layer and the output layer. The neural network is trained by the stochastic gradient descent algorithm, and the iterative weight and bias value are updated continuously to train the neural network. Generating output as a combination between input variables. Suppose we having n samples $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, we define the cost function as:

$$J(W, b) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{1}{2} \left\| f_{w,b}(x^i) - y^i \right\| \right) + \frac{\lambda}{2} \sum_{l=2}^{n_l} \sum_{j=1}^{S_{l-1}} \sum_{i=1}^{S_l} (W_{ij}^l)^2 \tag{6}$$

Where W and b are neural network parameters, W_{ij}^l represents the weight of the jth neuron from the l - 1th layer to the ith neuron of the lth layer, l=1,2,.... n_l, indicating the number layer , j = 1,2,... S_{l-1}, i = 1,2,..., S_l indicates the neuron is which layer of the network. The first term represents the mean error, the second part is the regular term, which is used to correct the weight to prevent over-fitting; $f_{w,b}(x^i)$ is a nonlinear hypothesis, which is defined as:

$$f_{w,b}(x) = \sigma(w^T x + b) \tag{7}$$

Where σ is the excitation function, and the sigmoid and RELU functions can be selected. The iterative update weight bias value formula is as follows:

$$\mathrm{W}_{\mathrm{ij}}^{\mathrm{l}} = \mathrm{W}_{\mathrm{ij}}^{\mathrm{l}} - \alpha \frac{\partial}{\partial \mathrm{W}_{\mathrm{ij}}^{\mathrm{l}}} \mathrm{J}(\mathrm{W}, \mathrm{b}) \tag{8}$$

$$\mathrm{b}_{\mathrm{i}}^{\mathrm{l}} = \mathrm{b}_{\mathrm{i}}^{\mathrm{l}} - \alpha \frac{\partial}{\partial \mathrm{b}_{\mathrm{i}}^{\mathrm{l}}} \mathrm{J}(\mathrm{W}, \mathrm{b}) \tag{9}$$

Where α is a learning factor

In this paper, we propose a multi-classifier ensemble method based on deep learning with k-fold cross-validation technology. As shown in Fig. 1, suppose k=4, 4 fold cross-validation, first divide the data set into 4 data sets, select k-1 sub-data sets as the training set, and the remaining 1 as the test set. The new data set is obtained by the multi-classifier as a training set of the deep neural network, the specific implementation steps are as follows, and Fig.1 is multi-classifier ensemble based deep learning model.

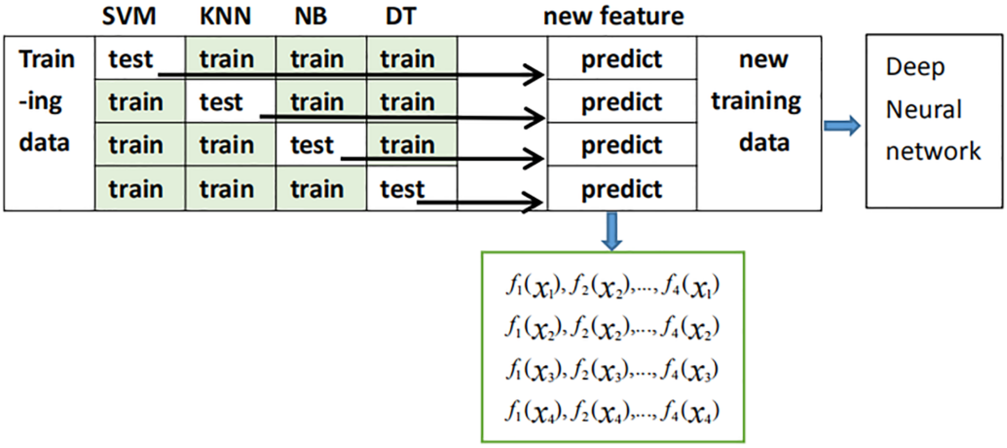

Figure 1. Multi-classifier ensemble based deep learning model

Step 1: Initialize the cross-validation coefficient k, and randomly divide the data set into k subsets $d_1, d_2, \ldots, d_k$,where $d_i = \{x_i, y_i\}$, $i = 1,2, \ldots, k$, we adopt k=4 in this paper.

Step 2: As shown in Fig.1, k-1 data sets are respectively used as training sets, and SVM, DT, NB, and KNN classifier models are trained, and the remaining one is used as a test set.

Step 3: Input x_i to each classification model, and get predictive value$f_1(x_i), f_2(x_i), \ldots, f_k(x_i)$, suppose$F_i = f_1(x_i), f_2(x_i), \ldots, f_k(x_i)$ where F_i is a new eigenvalue value of 0 and 1, combining F_i and y_i to new data set $d_i' = \{F_i, y_i\}$.

Step 4: Repeat step 3, we obtain a new subset of data $d_1', d_2', \ldots, d_k'$.

Step 5: The new data subset is also used in the deep neural network algorithm by cross-validation technology, and the obtained y_i' is compared with y_i, and calculate the final detection accuracy.

3. Results and Discussion

3.1 Dataset Source

The experimental data is mainly from the KDD Cup99 dataset, which contains four types of attacks: Dos (Denial of Service Attack), Probe (Scan and Detected Vulnerability), U2R (without permission granted by local users), R2L (unauthorized remote access attack). Currently, this data is still a recognized network intrusion detection data set. Each data contains 41 attributes and 1 label, of which 1~9 are the basic characteristics of the network connection, 10~22 is the content feature of the network connection, 23~41 is the traffic characteristic attribute, and the last label marks whether for normal data. The data used in this paper is shown in Table 1.

Table 1. Data distribution

Attack type	Training Set	Testing Set
DOS	11743	2299
U2R	52	20
R2L	1126	839
PROBE	4107	1208
NORMAL	14591	5783

3.2 Data Preprocessing

The data set has 41 attributes per data, including 3 symbol attributes and 38 numerical attributes. In order to be recognized by the algorithm, data is needed to preprocessing before detection.

(1) Character data conversion

As for flag type data, it can be converted to rej-0, rsto-2, ... , other-7, etc. For protocol type data, it can be converted into values: tcp-1, icmp-2, and so on. The same apply to the service attribute.

(2) Data normalization

Data normalization is a basic work of data mining. Because the dimensions and units used in data collection are different, the value range of data often varies greatly, and it is prone to large data eat the small. In order to avoid such a situation, this paper mainly adopts the maximum and minimum algorithm to normalize the data, and its calculation formula is as follows:

$$\mathrm{y} = \frac{x-min}{max-min} \tag{10}$$

Where max, min are the maximum and minimum values of the sample data, respectively.

3.3 Result Analysis

This experiment is mainly implemented in Python language and runs on the window 7 operating system. Using random forest feature selection algorithm to select the optimal feature subset, and then the data set without redundant features is used for the multi-classifier ensemble method based on deep learning training, compared with majority voting algorithm to verify whether the proposed method has higher accuracy.

(1) Setting the neural network parameters. This paper mainly uses a five-layer neural network. There is only one output layer, corresponding to 0 and 1, respectively. The weight of the network is initialized by He [15], the threshold is initialized to 0, and the learning rate sets to 0.01, iterate 1000 times, the coefficient of the regular term λ=3.

(2) Testing the performance of the network. Classifying the experimental results, and comparing with the expected values to calculate the detection accuracy.

In training set, the normal label is marked as 0, and the rest of the label is 1. The processed data is used in the designed training model. In this paper, the multi-classifier ensemble mainly includes support vector machine, Bayesian, decision tree and k-nearest neighbor four classifiers. The experimental results are shown in Table 2.

Table 2. Accuracy of detection

Alg	ACC(%)	Recall(%)
CFS+SVM	96.15	0.96
CFS+NB	93.54	0.94
CFS+DT	98.15	0.98
CFS+KNN	97.94	0.98
RF+SVM	95.25	0.95
RF+NB	91.07	0.91
RF+DT	99.10	0.99
RF+KNN	98.96	0.99
RF+MV	99.14	0.99
RF+Proposed	99.83	1.00

As showed in table 2, the random forest feature selection algorithm can significantly improve the performance of a single classification algorithm. In the decision tree classification, the accuracy of the RF+DT combination is up to 99.10%; for the KNN classifier RF+KNN also reached 98.90%; in these two classifiers, the decision tree's classifier works best. Compared with the correlation feature selection algorithm, the random forest feature selection algorithm removes noise more significantly and accurately. Increased by 0.095% and 1.02%, respectively. For the naive Bayesian algorithm, when there is an association between attributes, or the distribution is not uniform, it will lead to classification errors, so the detection accuracy is far lower than other classification algorithms. In the multi-classifier ensemble method, based on deep learning multi-classifier ensemble method has a detection accuracy of 99.83% and a recall rate of 1.00, which is 0.71% higher than majoring voting. The majoring voting ensemble method learns the linear relationship between multiple classification algorithms; As for deep learning ensemble method, it can automatically learn more complex nonlinear relationships between multiple classification algorithms, fully Take advantage of the data sets provided and ensure that the classification is more accurate.

4. Conclusion

The intrusion detection system is aiming at deal with the problem that the redundancy feature and classification algorithm causing the degradation accuracy of detection, in this paper, we propose a random forest feature selection and deep learning intrusion detection approach, and compare with the correlation feature selection algorithm and the majoring voting algorithm multi-model ensemble method through the experiment, the experimental results show that through the random forest feature selection algorithm to extract optimal feature subset, and then deep learning is employed to ensemble the output of multi-classifier, can effectively improve the accuracy of intrusion detection.

Acknowledgments

This word is supported by the science and technology project of Guangdong Province (No.2017B090906003) and the project of Guangzhou Science and Technology (No.201802010043, 2018 07010058).

References

[1]. Martin B, Rossouw V S. Utilizing fuzzy logic and trend analysis for effective intrusion detection[J]. Computers and Security,2003,22(5):423-434.

[2]. L Cong, Y Renwu and Z Changshui, Network intrusion detection based on FAST feature selection and ABQGSA-SVM [J]. Application Research of Computers, 2017, 34(7):2172-2179.

[3]. Sun N Q, LIY, Intrusion detection based on back-propagation neural network and feature selection mechanism [C] //Proc of FGIT,2009:151-159.

[4]. MU Qi, GONG Shangfu. Network Intrusion Feature Selection Based on Fast Attribute Reduction[J]. Computer Engineering,2011,37(17):113-115.

[5]. Bengio Y. Learning deep architectures for AI [J]. Foundations and Trends in Machine Learning,2009,2(1):1-127.

[6]. Denning DE, An intrusion detection model- [J]. IEEE Transaction Software Engineering, 2003,13(2):222-232.

[7]. CHEN You, CHENG XueQi. Lightweight Intrusion Detection System Based on Feature Selection[J]. Journal of Soft-ware,2007,18(7):1639-1651.

[8]. L Cong and Y Renwu, Feature selection and SVM parameter optimization based on IBQGSA in intrusion detection[J]. Computer Engineering and Design, 2017, 38(8): 2227-2234.

[9]. Gong An, Lv Qian and Hu Changjun. Parameter Optimization and Application of SVM Based on Chaos Gravitational Search Algorithm[J]. Computer Science, 2015, 42(4):240-243.

[10]. XIAO Guorong. Network intrusion detection by combination of improved ACO and SVM. Computer Engineering and Applications, 2014, 50(3):75-78.

[11]. Hu Mingxia. Intrusion Detection Algorithm Based on BP Neural Network [J]. Computer Engineering, 2012,38(6):148-150.

[12]. Shen Xiajiong, Wang Long, Han Daojun. Application of BP Neural Network Optimized by Artificial Bee Colony in Intrusion Detection[J]. Computer Engineering,2016,42(2): 190-194.

[13]. Ma H. Correlation-based feature selection for machine learning [D]. Hamilton: The University of Waikato,2000.

[14]. Janitza S, Strobl C, Boulesteix A L. An AUC-based permutation variable importance measure for random forests [J]. BMC bioinformatics, 2013, 14(1): 119.

[15]. He K, Zhang X, Ren S, et al. Delving deep into rectifiers: Surpassing human-level performance on image net classification [C] //Proceedings of the IEEE international conference on computer vision. 2015: 1026-1034.