A Hybrid Feature Selection Method for Network Traffic Anomaly Detection

Haoming Wu¹², Bin Zhang¹², Shuqin Dong¹²
¹ PLA SSF Information Engineering University, Zhengzhou, China
² Henan Key Laboratory of Information Security, Zhengzhou, China
wuhaoming0512@126.com

Abstract. In order to keep fast and accurate in feature selection for network traffic anomaly detection, this paper proposes a hybrid feature selection method. Firstly, to reduce the calculation and to identify the redundant features, we regard the ratio of mutual information between features to a feature entropy as the redundancy degree of the feature. If the ratio is greater than a predefined threshold, the feature is judged as redundant and will be deleted from the feature set. Secondly, based on the feature set whose redundant features have been removed, this method uses the ratio of the anomaly detection accuracy after and before delete one feature from the feature set to measure the effect of the feature on detection. Then, the features are sorted in ascending order of the ratio and the top k features with the highest detection accuracy are selected as the result. Experimental results show that the proposed method can quickly screen out a feature subset with good detection performance and lower dimensions.

1. Introduction
Massive and high-dimensional data are a major problem in network traffic anomaly detection, which seriously affects the efficiency of anomaly detection. Feature selection, as an effective way of dimensionality reduction [1], can select a subset from the original feature set to reduce the time consumption of anomaly detection, increase the detection accuracy, and improve the interpretability of anomaly detection models [2]. It’s one of the effective ways to solve the above problem. According to the feature selection process depends on the anomaly detection classifier or not, the feature selection methods can be divided into the filter and the wrapper [3]. The filter methods don’t rely on the classifier to measure the importance of features. It identifies the important features according to their attribute. In general, the filter methods are very efficient, and have high generalization ability, but the selected feature subset often cannot achieve great anomaly detection performance. The typical algorithms of the filter methods include ReliefF [4] algorithm, mRMR (Maximum Relevance Minimum Redundancy) [5] algorithm, and GRM (Global Redundancy Minimization) [6] algorithm, etc. In contrast, the feature selection process of the wrapper methods depends on the classifier. It employs the anomaly detection performance to measure feature subset. This kind of methods are usually combined with GA (Genetic Algorithm) [7], EM (Electromagnetism-like Mechanism) [8] algorithm, PSO (Particle Swarm Optimization) [9] algorithm and other random search algorithms to find the optimal feature subset. The wrapper methods are relatively simple to realize and could yield a better result than the filter methods, but it is more time consuming and has less generalization ability.
Recently, researchers developed several hybrid feature selection methods which are the combination of the filter and the wrapper to obtain the advantage of both methods [10]. The common hybrid methods can be divided into two phases. It first employs a filter method to select a smaller feature subset, which aims to decrease the computation cost of the second stage, and then utilizes a wrapper method to search the best result. For example, [11] combines the IG (Information Gain) algorithm with the BPSO (Binary Particle Swarm Optimization) algorithm; [12] combines the mRMR algorithm with the GA; [13] combines three filter methods, the Chi-square, the IG and the GR(Gain Ratio) algorithm, with the SFS(Sequential Forward Selection) algorithm. However, these methods just utilize the filter followed by the wrapper. The correlation between these two phases is weak, and the relationship between the features is split. Moreover, the number of features selected by the filter methods is something difficult to set. It is likely to fall into the local optimum with a small value, and the larger make it hard to reduce the time consumption.

Aiming at the above problems, this paper proposes a hybrid feature selection algorithm which can be divided into two phase. In the first stage, we propose a redundant feature deletion mechanism based on information theory to identify and delete redundant features. A feature importance evaluation mechanism is proposed in the second stage based on the wrapper methods. The features that remain in the feature set after the first stage are sorted according to their importance, and the top k features with the highest anomaly detection accuracy are selected as the result of the algorithm.

2. Methodology

For a dataset $D = \{X_1, X_2, \ldots, X_d, Y\}$, where $X_i = \{x_{i1}, x_{i2}, \ldots, x_{in}\}$ is the $i$th feature which consist of the $i$th feature value of each record, $Y = \{y_1, y_2, \ldots, y_n\}$ is the class label, $d$ is the number of features, and $n$ is the number of records, the effect of the feature $X_i$ on anomaly detection can be considered from two aspects, i.e., the direct effect on anomaly detection of $X_i$, and the influence of the relationship between $X_i$ and other features on anomaly detection. If we delete $X_i$ from $D$, the effect of the feature on anomaly detection in both aspects can be removed simultaneously. Comparing the anomaly detection accuracy after and before deleting $X_i$ from $D$, it is possible to measure the effect of $X_i$ on the anomaly detection in both aspects simultaneously.

However, there are two issues in the method above. Firstly, considering such a situation, there exists a feature $X_j (j = 1, 2, \ldots, d, j \neq i)$ which is a redundant feature of $X_i$. If we delete $X_i$ from the dataset, the information that the dataset contains doesn’t change due to the existence of $X_j$, so that the change of the anomaly detection accuracy before and after deleting $X_i$ from the dataset can’t reflect the effect of $X_i$ precisely. And the second problem is that the method above needs to compute the anomaly detection accuracy $d$ times when there are $d$ features in the dataset, which would be time consuming. Therefore, a mechanism to remove potentially redundant features before employing this method is needed to solve the problems above.

2.1. Redundant features deletion mechanism

In general, the information theory-based feature selection methods use mutual information to identify redundant features. For instance, [14] utilizes (1) to select the feature which is high relevant to the class $Y$ and low redundant to the selected features.

$$J(X_i) = I(X_i; Y) - \beta \sum_{X_s \in S} I(X_s; X_i)$$

(1)

Where $X_i$ is the candidate feature, $S$ represent the selected features, $X_i$ belongs to $S$, $\beta$ is the weight of redundancy which usually set to $|S|^{-1}$, $I(X_i; Y)$ is the mutual information between feature $X_i$ and class $Y$, and $I(X_s; X_i)$ is the mutual information between $X_s$ and $X_i$. $\beta \sum_{X_s \in S} I(X_s; X_i)$ in (1)
represents the redundancy between \( X_i \) and \( S \). If \(|S| = 1\), \( \beta \sum_{X_i \in S} I(X_i; X_j) = I(X_i; X_j) \). It means that the mutual information between feature \( X_i \) and \( X_j \) can be used to measure the redundancy among them. Nevertheless, \( I(X_i; X_j) \) cannot represent the redundancy degree directly. According to the information theory, mutual information can be calculated using (2).

\[
I(X_i; X_j) = H(X_i) - H(X_i | X_j)
\]

where \( H(X_i) \) is the entropy of \( X_i \), and \( H(X_i | X_j) \) is the conditional entropy of \( X_i \) given \( X_j \).

Obviously, given \( X_j \), if \( H(X_i | X_j) > I(X_i; X_j) \), \( X_j \) could provide much more information even though \( I(X_i; X_j) \) is large. Hence, the redundancy degree is defined as (3).

\[
R_{r,s} = \frac{I(X_i; X_j)}{|H(X_i)|^{-1}}
\]

where \( R_{r,s} \) is the redundancy degree of \( X_i \) relative to \( X_j \), and \( R_{r,s} \in [0, 1] \). \( R_{r,s} \) close to 1 means that \( X_j \) contains most of the information of \( X_i \) and \( X_j \) can be removed from the feature set. But considering a situation that \( R_{r,s} \) is also close to 1, it denotes that \( X_j \) and \( X_s \) are redundant to each other. In such a case, the feature whose redundancy degree is larger should be removed.

However, the condition that \( R_{r,s} \) close to 1 cannot use to identify the redundant features in practice. A threshold is needed to determine \( R_{r,s} \) close to 1 or not. If \( a \) is used to represent the threshold, the description of the redundant feature deletion mechanism is as follow: \( \forall X_s, X_i \in D \) \((i \neq s)\), if \( R_{r,s} \geq a \) and \( R_{r,j} \geq R_{r,j} \), the feature \( X_i \) is recognized as redundant feature and should be removed from \( D \).

2.2. Feature importance evaluation mechanism

Let \( D_r = \{X_1, X_2, \ldots, X_h, Y\} \) be the dataset that has removed the redundant feature. \( h \) is the number of features in \( D_r \), and \( X_p \) \((p = 1, 2, \ldots, h)\) is the \( p \)th feature of \( D_r \). The importance of each feature in \( D_r \) can be calculated as following ways.

Firstly, \( D_r \) is used for anomaly detection and the detection accuracy is recorded as \( ACC_{D_r} \). Secondly, we remove \( X_p \) from \( D_r \) and get \( D'_r = \{X_1, X_2, \ldots, X_{p-1}, X_{p+1}, \ldots, X_h, Y\} \). The anomaly detection accuracy \( ACC_{D'_r} \) is calculated based on \( D'_r \). Then, \( ACC_{D'_r} \) is compared with \( ACC_{D_r} \). If \( ACC_{D_r} > ACC_{D'_r} \), namely, the accuracy has dropped after deleting \( X_p \). It means that \( X_p \) contains important information and could promotes the anomaly detection. On the contrary, if \( ACC_{D_r} \leq ACC_{D'_r} \), \( X_p \) is unnecessary or adverse for anomaly detection. Hence, (4) is provided to measure the importance of \( X_p \). \( Score_p \) is inversely proportional to the feature importance.

\[
Score_p = \frac{ACC_{D'_r}}{ACC_{D_r}} \times (ACC_{D_r})^{-1}
\]

According to (4), \( Score_p \) of each feature can be calculated and the features in \( D_r \) are rearranged in ascending order of \( Score_p \). We use \( D_{as} = \{X_1, X_2, \ldots, X_h, Y\} \) to represent the rearranged dataset, and \( a_1, a_2, \ldots, a_h \) is the order of features after rearranged. To select the important features and obtain the best performance of anomaly detection, the top \( k(k = 1, 2, \ldots, h) \) features in \( D_{as} \) is chosen to form \( D_k = \{X_{a_1}, X_{a_2}, \ldots, X_{a_k}, Y\} \) and the anomaly detection accuracy based on \( D_k \) is calculated and recorded as \( ACC_{D_k} \). The maximum value \( ACC_{D_{as}} \) is \( \{ACC_{D_k}(k = 1, 2, \ldots, h)\} \) is find out, and the feature subset \( \{X_{a_1}, X_{a_2}, \ldots, X_{a_m}\} \) is returned as the result.

The algorithm pseudocode is as follow:
**Input:** dataset $D$, anomaly detection classifier $C$

**Output:** optimized feature subset $S$

1) **FOR** $i = 1$ **TO** $d$

2) **FOR** $j = 1$ **TO** $d$

3) \[ R_{i,j} = I(X_i;X_j)[H(X_i)]^{-1} \quad \text{//Computing the redundancy degree between any two features} \]

4) **END FOR**

5) **END FOR**

6) $D_r = D$

7) **FOR** $i = 1$ **TO** $d$

8) **FOR** $j = 1$ **TO** $d$

9) \[ \text{IF} \quad R_{i,j} > \alpha \quad \& \& \quad R_{j,i} \geq R_{i,j} \quad \& \& \quad i \neq j \quad \& \& \quad X_i, X_j \in D_r \]

10) \[ D_r = D_r - \{ X_i \} \quad \text{//Deleting the redundant features} \]

11) **END IF**

12) **END FOR**

13) **END FOR**

14) Using $D_r$ to calculate the anomaly detection accuracy $ACC_{D_r}$

15) **FOR** $p = 1$ **TO** $h$

16) \[ D_r^p = D_r - \{ X_p \} \quad \text{//deleting one feature from $D_r$} \]

17) Using $D_r^p$ to calculate the anomaly detection accuracy $ACC_{D_r^p}$

18) \[ \text{Score}_p = ACC_{D_r^p} (ACC_{D_r^p})^{-1} \quad \text{//Calculating the importance score of $X_p$} \]

19) **END FOR**

20) $D_r = D_r.\text{Sort()} \quad \text{//the features in $D_r$ is sorted in ascending order of $\text{Score}_p$}$

21) **FOR** $k = 1$ **TO** $h$

22) Choosing top $k$ features of $D_r$ to form $D_k$

23) Using $D_k$ to calculate the anomaly detection accuracy $ACC_{D_k}$

24) **END FOR**

25) \[ S = \arg \max_{\alpha} \{ ACC_{D_{\alpha}} \} (k=1,2,...,h) \]

26) **RETURN** $S$

---

### 3. Experiment

Several experiments were carried out to demonstrate the efficiency of our algorithm. The experimental environment is a portable computer with 64-bit Windows 10 operating system, 16GB RAM and Intel(R) Core(TM) i7-7700HQ 2.80GHz processor. The algorithm is implemented on matlab, SVM is chosen as the anomaly detection classifier, and the NSL-KDD [15] which contains 41 features and 5 main categories records is used as the benchmark dataset. We select the ‘KDDTrain+ 20Percent’ and ‘KDDTest+’ from NSL-KDD as the training set and the testing set respectively.

3.1. Redundancy degree threshold
The redundancy degree threshold \( a \) is a key parameter of our algorithm which directly affects the performance of our algorithm. To get an appropriate value of \( a \) which could maintain the performance of anomaly detection and minimizing the time consumption of the algorithm, different value of \( a \) is employed to test our algorithm. Figure 1 presents the line graph of the anomaly detection accuracy comparison with different number of selected top features under different value of \( a \). ‘OS’ represents that only the second stage of our algorithm is used, namely, we do not delete feature in the first stage.

From Figure 1, it is obvious that the curves of \( a \geq 0.9 \) are similar to the OS curve. The maximum detection accuracy of \( a \geq 0.9 \) is 72.26%, 72.28%, and 72.41% respectively which is close to 72.47% , the maximum value of OS. When \( a < 0.9 \), the fluctuation of the curves is large and the performance is unstable even though the maximum value of the anomaly detection accuracy is close to 72%. Limited by the space, the curves of the false positive rate (FPR, the percentage of the normal records wrongly flag as anomaly) and the false negative rate (FNR, the percentage of the anomaly records wrongly flag as normal) are not provided. It is similar to the curves of detection accuracy. In addition, the smaller the number of features selected in the first phase, the less time computation is required for the second phase. The threshold \( a \) is set to 0.9 in the experiment.

### 3.2. Algorithms Performance Analysis

To verify the effectiveness of the proposed algorithm, our algorithm (we use OA to represent our algorithm in the following) is compared with ReliefF, mRMR, GA, ReliefF+GA, mRMR+GA, HFWFS [16]. ReliefF+GA represents the hybrid method that ReliefF is used first and GA is run based on the feature subset selected by ReliefF, and mRMR+GA is same as ReliefF+GA. In addition, we also compare OA with OS which can be described as OA without the first stage to demonstrate the effects of the two mechanisms.

The parameters of the algorithms above are set as follows. The sampling rate and the number of the nearest neighbors of ReliefF are set to 0.2 and 5 respectively, and the top \( k_1 \) features that have the highest detection accuracy are selected. mRMR algorithm selects the top \( k_2 \) features with the highest detection accuracy. As for the GA, the size of the population is set to 100, the maximum iteration is set to 100, the crossover rate is 0.6, the mutation rate is 0.01, the fitness function is the detection accuracy of SVM, and the feature selection result is the individual with the highest classification accuracy in the 100 iterations. The number of features selected by ReliefF in ReliefF+GA are set to 29 (the number of features selected by OA in the first stage), and mRMR selects 16 (i.e. \( k_2 \)) and 29 features respectively in mRMR+GA. For HFWFS, the number of features selected by Chi-square, IG and GR is 22, 12 and 9 respectively. It is the same as the parameter value set in [13].

#### 3.2.1. Feature selection result

The time consumption and the number of features selected by different algorithms are shown in Table 1. In the algorithm column of table 1, the number in the parentheses of ReliefF+GA and mRMR+GA represents the number of features selected by ReliefF and mRMR (e.g. ReliefF(29)+GA means that the number of features selected by ReliefF in the hybrid method is 29). The features of NSL-KDD are represented by 1 to 41 in table 1, according to the default order of features. If more information about the features is required, please refer to [17].
Table 1. The time consuming and the number of selected features of different algorithms.

| Algorithm   | Time consuming/s | Number of selected feature | Selected feature                                                                 |
|------------|------------------|----------------------------|-----------------------------------------------------------------------------------|
| OA         | 685.47           | 12                         | 30,36,40,34,32,33,37,31,24,39,25,3                                                |
| OS         | 920.45           | 22                         | 30,36,37,31,40,23,27,41,26,6,7,19,22,28,10,13,1,9,11,14,15                        |
| RelieF     | 35400.42         | 40                         | 29,38,12,32,36,34,33,2,4,23,3,39,35,40,30,26,31,8,27,37,22,41,24,28,5,14,15,10,11,16,13,6,17,18,20,21,5 |
| mRMR       | 402.52           | 16                         | 29,35,38,12,33,25,23,39,37,26,36,34,32,3,41                                     |
| GA         | 106583.87        | 22                         | 1,3,5,9,11,12,13,14,15,18,19,20,22,28,30,31,32,34,36,37,38,39                   |
| ReliefF(29)+GA | 89036.95       | 18                         | 3,10,12,14,15,25,26,27,28,29,33,34,35,36,37,39,40,41                         |
| mRMR(16)+GA | 79427.99       | 11                         | 3,4,12,29,31,33,34,36,37,39                                                     |
| mRMR(29)+GA | 93572.13       | 15                         | 3,8,10,14,17,22,25,27,30,31,32,34,36,37,41                                     |
| HFWFS      | 642.14           | 2                          | 30,5                                                                            |

As can be seen from the table, compared with OS, the time consumption and the number of features selected by OA are reduced by 234.98 seconds and 10 respectively. It means that the redundant feature deletion mechanism can effectively reduce the time cost of the algorithm and decrease the number of selected features. Compared with other algorithms, the time consumption of OA is 1.70 times of mRMR and 1.07 times of HFWFS, and less than 2% of other algorithms. The number of features selected by OA is 1 and 10 higher than that of mRMR(16)+GA algorithm and HFWFS algorithm respectively, and lower than the number of features selected by other algorithms.

3.2.2. Anomaly detection performance. As show in Figure 2, the accuracy, the false positive rate, and the false negative rate of anomaly detection are used to measure the performance of the feature subset selected by different algorithms.

Figure 2 (a) presents the comparison of the detection accuracy. Compared with All Feature, OS, ReliefF, mRMR, GA, ReliefF(29)+GA, mRMR(16)+GA, mRMR(29)+GA, and HFWFS, the detection accuracy of OA is changed by 3.15%, -0.21%, 3.14%, 2.67%, -2.06%, 0.06%, -3.53% respectively (the negative numbers indicate that the accuracy of OA is lower than the corresponding algorithm). The accuracy of OA is higher than All feature, ReliefF, mRMR, ReliefF(29)+GA, mRMR(16)+GA and HFWFS.

Figure 2 (b) illustrates the false positive rate of anomaly detection. In Figure 2 (b), we can see the FPR of OA is changed by -0.43%, 0.30%, -0.43%, -0.83%, 4.33%, 3.60%, 2.65%, 3.87%, and 3.00% respectively compared with other algorithms. The FPR of OA is lower than All feature, ReliefF, and mRMR. In Figure 2 (c), the FNR of OA is changed by -5.66%, -0.25%, -5.61%, -3.75%, 1.21%, 1.49%, -2.09%, 0.72%, and -9.49% respectively compared with other algorithms. The FNR of OA is lower than All feature, OS, ReliefF, mRMR, mRMR(16)+GA and HFWFS.

From the above, the performance of OA is better than the filter method and without feature selection, and the detection accuracy and the FNR are similar to the other hybrid methods. But the FPR of OA is worse than the hybrid method. This is because OA only uses classification accuracy as the feature selection criteria. OA improves the classification accuracy of the four kinds of anomaly records at the cost of decreasing the classification accuracy of the normal records.

4. Summary
In order to quickly and accurately select the feature subset that has a great performance on network traffic anomaly detection and has removed the useless and interference features, this paper proposes a hybrid feature selection method. The method comprehensively considers the effect of features on anomaly detection, can accurately measure the importance of features and quickly select feature subsets with better performance, and effectively reduces the dimension of the original feature set. However, the FPR of the feature subset selected by the method is worse than the other hybrid method. We will continue to improve the performance of the algorithm in the future.

5. References

[1] Chenping H, Feiping N, Xuelong L, Dongyun Y, and Yi W 2014 Joint Embedding Learning and Sparse Regression: A Framework for Unsupervised Feature Selection. J. IEEE T CYBERNETICS. 44 793-804.

[2] AHMAD A S, TAGHI M K, and RANDALL W 2014 Proc. BIBE ’14(Boca Ratón, USA) Evaluation of Wrapper-based Feature Selection using Hard, Moderate, and Easy Bioinformatics Data. 149-55.

[3] Xiaonian W, Xiaojin P, Yuyang Y, and Kun F 2015 Two-level Feature Selection Method Based on SVM for Intrusion Detection. J. Journal on Communications. 36 19-26.

[4] Palma-Mendoza R J, Rodriguez D, and De-Marcos L 2018 Distributed ReliefF-based Feature Selection in Spark. J. KNOWL INF SYST. 57 1-20.

[5] Peng H, Long F, and Ding C 2005 Feature Selection Based On Mutual Information: Criteria of Max-Dependancy, Max-Relevancy, and Min-Redundancy. J. IEEE T PATTERN ANAL. 27 1226-38.

[6] Wang D, Nie F, and Huang H 2015 Feature Selection via Global Redundancy Minimization. J. IEEE T KNOWL DATA EN. 27 2743-55.

[7] Huang C L, and Wang C J 2006 A GA-based Feature Selection and Parameters Optimization for Support Vector Machines. J. EXPERT SYST APPL. 31 231-40.

[8] Su C T, and Lin H C 2011 Applying Electromagnetism-like Mechanism for Feature Selection. J. INFORM SCIENCES. 181 972-86.

[9] Chuang L Y, Chang H W, Tu C, and Yang C 2008 Improved Binary PSO for Feature Selection using Gene Expression Data. J. COMPUT BIOL CHEM. 32 29-38.

[10] Alzubi R, Ramzan N, Alzoubi H, and Amira A 2017 A Hybrid Feature Selection Method for Complex Diseases SNPs. J. IEEE ACCESS. PP(99):1-1.

[11] Zhao X, Li D, Yang B, Huiling C, Xinbin Y, Chenglong Y, and Shuangyin L 2015 A Two-stage Feature Selection Method with its Application. J. COMPUT ELECTR ENG. 47 114-25.

[12] Akadi A E, Amine A, Ouardighi A E, and Aboutajidine D 2011 A Two-stage Gene Selection Scheme utilizing MRMR Filter and GA Wrapper. J. KNOWL INF SYST. 26 487-500.

[13] Belouch M, Elhadaj S, and Idhammad M A Hybrid Filter-Wrapper Feature Selection Method for DDoS Detection in Cloud Computing. J. INTELL DATA ANAL. 22 1209-26.

[14] HongFang Z, Yao Z, Yingjie Z, and Hongjiang L 2018 Feature Selection Based on Conditional Mutual Information: Minimum Conditional Relevance and Minimum Conditional Redundancy. J. APPL INTELL. 49 883-96.

[15] Canadian Institute for Cybersecurity 2018 The NSL-KDD dataset. Available: https://www.unb.ca/cic/datasets/nsl.html.

[16] Belouch M, Elhadaj S, and Idhammad M L 2018 A Hybrid Filter-Wrapper Feature Selection Method for DDoS Detection in Cloud Computing. J. INTELL DATA ANAL. 22 1209-26.

[17] Yallei D and Yuqing Z 2018 Proc. CSAI ’18 (Shenzhen, China) Intrusion Detection System for NSL-KDD Dataset Using Convolutional Neural Networks. 81-85.

Acknowledgments

This paper is supported by the Foundation and Frontier Technology Research Project of Henan Province (ID:142300413201).