An Unknown Attack Detection Scheme Based on Semi-supervised Learning and Information Gain Ratio

Meng-Fan Xu, Xing-Hua Li, Mei-Xia Miao, Cheng Zhong, Jian-Feng Ma

School of Cyber Engineering, Xidian University, China
812455541@qq.com, xhli1@mail.xidian.edu.cn, miaofeng415@163.com,
240324665@qq.com, jfma@mail.xidian.edu.cn

Abstract

State-of-the-art intrusion detection schemes employ machine learning techniques to identify unknown attacks with the network traffic data features. However, due to the lack of enough training set, the difficulty of quantitatively and adaptively selecting features, the existing schemes cannot detect unknown attacks effectively. To address this issue, this paper first proposes an improved k-means driven semi-supervised learning algorithm to enlarge the training set accurately with a small amount of labelled dataset for the detection model. Furthermore, information gain ratio aware random forest is utilized to determine the impact of different features and their weight voting for determination of unknown attacks, which can not only retain the information of features at utmost, but also adjust the weights of different features adaptively against dynamic attacks. Extensive experiments indicate that this scheme can detect unknown attacks effectively with more than 91% accuracy and less than 5% false negative rate over three real-world datasets. Compared with existing schemes, the accuracy is increased by at least 15.85%, while the false negative rate is decreased by more than 51.98%.

Keywords: Unknown attack detection, Feature selection, Semi-supervised, Information gain ratio

1 Introduction

With the rapid development of network-based computer services and applications, the security problems have become increasingly prominent.

The host/network security, especially in Internet of things or industrial networks, becomes difficult to achieve. The Intrusion Detection System (IDS) has been widely used to monitor and analyze host or network events, and it can be used to identify the deviations from normal host/network behaviors. The existing IDS can be divided into two categories: misuse-based IDS and anomaly-based IDS [1]. The misuse-based IDS can detect known attacks effectively, such as Snort [2], which is the most famous open source system. This type of IDS has a low false alarm rate for detecting the known attacks, but it cannot identify new or unknown attacks. The anomaly-based IDS identify the intrusion behavior by building a model of normal host/network behavior and then the behavior with any significant deviation from the model is identified as an intrusion. This type of IDS can detect new or unknown attacks, but with high false positive and false negative rates. In order to detect unknown attacks effectively, machine learning algorithms have been vastly adopted in the literature. In particular, the network traffic features of the target network are selected using different feature selection methods and the attacks are detected by the supervised learning algorithm [3-14]. Existing anomaly-based IDSs using machine learning algorithms will select a subset of features to train a model which is further utilized to identify unknown attacks. However, they still face following three challenges.

The large scale training set is difficult to obtain. They are often artificially generated by some experts (*i.e.*, to decide whether an instance of data is an unknown attack or not). However, existing anomaly-based IDS methods, which utilize machine learning, all rely upon a large amount of labelled training data. Lack of training samples inevitably limits the accuracy of detection task.

The appropriate traffic features for a given unknown attack are difficult to select quantitatively. However, existing attack detection methods use heuristic algorithms to search for a fixed subset of features according to some qualitative study, while ignoring some features which may affect the determination result significantly, leading to an inferior detection ability.

The important network traffic features of unknown attacks are difficult to select adaptively. The important features of network traffic for determination of various attacks are also different.

To address above problems, we propose an unknown attack detection scheme that utilizes semi-supervised learning technique and information gain ratio. Our

[*] Corresponding Author: Xing-Hua Li; Email: xhli1@mail.xidian.edu.cn

DOI: 10.3966/160792642019032002030

main contributions are summarized as follows.

- An improved k-means driven semi-supervised learning algorithm is proposed, with which the large amount of unlabeled data samples are labelled accurately. As a result, a large scale training set is obtained to improve the detection performance for unknown attacks.
- An information gain ratio aware random forest model is presented, which is utilized to determine the important features and their weights voting for determination of various unknown attacks quantitatively and adaptively.
- Extensive experiments indicate that our scheme can detect unknown attacks effectively with more than 91% accuracy and less than 5% false negative rate over three real-world datasets. Compared with existing schemes, the accuracy (Acc) is increased by at least 15.85%, while the false negative rate(FNR) is decreased by more than 51.98%.

The remainder of this paper is organized as follows. The related work is discussed in Section 2 and the scheme is presented in detail within Section 3. In Section 4, we conduct series of empirical studies over three real-world datasets. Finally, this work is concluded in Section 5.

2 Related Work

The non-machine-learning-based scheme, such as verifiable computation-based [15-17], cloud computing-based [18-21] scheme have become more popular for known attacks detection. Unfortunately, they are incapable of dealing with big data or detecting unknown attacks.

To deal with big data and unknown attacks, the machine learning algorithms are utilized to detect intrusion. The network traffic anomaly or host malicious behavior is utilized for attacks detection.

Ashfaq et al. [9] proposed an novel fuzziness based semi-supervised learning approach by utilizing unlabeled samples assisted with a supervised learning algorithm to improve the classifier's performance for the IDSs. The experimental results show that unlabeled samples belonging to low and high fuzziness groups make major contributions to improve the classifier's performance compared to existing classifiers. Al-Yaseen et al. [10] designed a model that deals with real intrusion detection problems in data analysis, and classify network data into normal and abnormal behaviors. They proposed a multi-level hybrid intrusion detection model that used support vector machine and extreme learning machine to improve the efficiency of detecting attacks. Viegas et al. [11] proposed and evaluated three new approaches to improve the energy efficiency of network security algorithms and applications. They presented detailed energy consumption measurements for all algorithms, in the aspects of both software and hardware. The new feature extractor consumes only 22% of the energy used by a commercial tool. Zhu et al. [12] proposed a scheme for the manyobjective problems in IDS, which used a special domination method and a predefined multiple targeted search for population evolution. It can differentiate traffic not only between normal and abnormal but also by abnormal type with both higher classification accuracy and lower computational complexity. Yan et al. [13] presented a novel system PeerClean that detected P2P botnets in real time using only high-level features extracted from C&C network traffic. To increase the detection probability, they further proposed to train the model with average group behavior, where the extreme group behaviors are explored for the detection. They reported high detection rates with few false positives. Jaswal et al. [22] proposed a hybrid machine learning algorithm following with rule generation algorithm to detect the intrusion in the network logs by training KDD dataset. Training and testing KDD data set provides the way of analyzing the actual behaviors and predicted behaviors of the network logs.

These researches rely on sufficient training set, such as the popular datasets KDD99 and NSL-KDD. When these schemes are directly used in the actual network, the detect model cannot be fully trained with the limited scale of accurate and labelled data samples, thereby unknown attacks cannot be identified effectively. Meanwhile, the important features of network traffic can be only selected through the expert knowledge qualitatively. The information contained in unselected features is completely ignored, which leads to the loss of effective features and the inferior ability for detecting unknown attacks.

3 Our Proposed Scheme

In this section, we present a variation of Random Forest, where trees are built according to information gain ratio and final labels are generated by weighted voting. In the following, we shall introduce both methods in sequence. The notations that will be used in the following discussion are summarized in Table 1.

3.1 Inferring the Labels of Unlabeled Training Samples via Semi-Supervised Learning

Due to the large amount of historical data in network traffic, it is difficult to label them purely using expert knowledge artificially, which prevents the detection model from performing accurately for unknown attack detection [23]. To address this issue, semi-supervised learning can be employed by using a small amount of prior knowledge to assist unsupervised learning [23-24]. In order to ensure that the historical data can be labelled accurately, an improved k-means driven semi-supervised learning method is proposed in this paper. Given a group of training samples (*i.e.*, $D_0 = \{N_1, ..., N_I\}$), which consist of a small number of labelled samples (*i.e.*, normal ones $L_0 \subset D_0$ and abnormal ones $L_1 \subset D_0$) and a large number of unlabeled ones, the learning method (shown in Figure 1) can be described as follows. It is formally shown in Algorithm 1.

Table 1. Summary of notations

Symbol	Instruction
N_i	All pairs of samples
c_j	The cluster centers
$d(N_i, c_j)$	The Euclidean distance between N_i and c_j
$N_{j,\ i}$	The samples in cluster c_i
n	The number of updates
I'	The number of samples in cluster c_j
J	The accumulated distance
$c(N_i)$	The centroid of the assigned cluster for data sample N_i
C_i	The final cluster
$l(C_i)$	The assigned label for all samples in C_i
S_q	The subset of data set D by using Bootstrap algorithm
$S^{(a)}_{v+}/S^{(a)}_{v-}$	The set of samples whose values on attribute a are greater (resp., ⩽) than v
a	The feature of data N_i
v	The value of feature a
$v_{OPT}(a)$	The corresponding split value
T_q	A decision tree

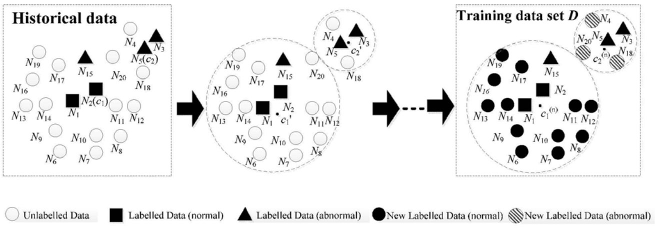


Figure 1. Inferring the labels of unlabeled training samples

Algorithm 1. Training samples labelling algorithm

```
Input: A large amount of historical data
Output: A training set D
1.  for the normal labelled samples do
2.      randomly select one sample as the initial cluster center
3.  end for
4.  for the abnormal labelled samples do
5.      randomly select one sample as the initial cluster center
6.  end for
7.  for each $N_i$ do
8.      calculate $d(N_i, c_j)$
9.  end for
10. for each unlabeled samples do
11.     assign it to the nearest cluster
12. end for
13. update the $c_j^{(n)}$, $c_j^{(n)} = \frac{\sum_{i=1}^{I'} N_{j,i}}{I'}$
14. repeat
15.     step 7 to 10
16.     calculate $J$, $J = \sum_{i=1}^{I} d(N_i, c(N_i))$
17. until $J$ over each data sample and its assigned cluster center reaches the minimum
18. calculate the $p(C_i \mid L_j)$, $p(C_i \mid L_j) = \frac{|L_j \cap C_i|}{L_j}$ ($j$=0, 1 and $i$=1, 2)
19. label all samples in $C_i$, $l(C_i) = \text{argmax}_i\{p(C_i|L_j)\}$ ($j$ = 0, 1)
20. return D
```

For example, I = 20 and I' = 15 in cluster 1 of training set D_0 as shown in Figure 1, where $L_0=\{N_1, N_2\}$, $L_1=\{N_3, N_4, N_5\}$ and $D_0=\{N_1, \cdots, N_{20}\}$. When our scheme ends (i.e. after step 4) with the minimal J, we finally get $C_1=\{N_1, N_2, N_6, \cdots, N_{17}, N_{19}\}$ and $C_2=\{N_3, \cdots, N_5, N_{18}, N_{20}\}$. Then we have to identify whether C_1 (resp. C_2) is the normal class. This is done by measuring the posterior probability of $p(C_i|L_j)$. Obviously, for C_1, $p(C_1|L_0) = 1$ and $p(C_1|L_1) = 0$, then C_1 is probably normal class according to step 5.

3.2 Information Gain Ratio Aware Random Forest

Intuitively, more information that a feature can bring to a classification model, more important the feature is. The presence of a feature in the model will necessarily lead to a change in the amount of information, which can be measured by information gain [25]. To select a more representative feature in the learning model, we adopt information gain ratio in this scheme for measuring the capability of candidate features to distinguish the class in the training data. Then we present a variation of random forest algorithm by embedding information gain ratio in feature selection for each independent tree. Formally, the concept of information gain and information gain ratio are illustrated as follows.

Definition 1: [Information Gain] Given a dataset $S = \{N_1, ..., N_I\}$ consisting of positive samples S^+ and negative ones S^-, where $N_i = \{N_{i1}, ..., N_{im}\}$ (m denotes the number of features), let $H(S)$ and $H(S|a)$ be the entropy of S and conditional entropy of S over attribute a, respectively, then the *Information Gain* of S over attribute a can be acquired by:

$$G(S, a) = H(S) - H(S|a). \quad \textbf{(1)}$$

In particular, entropy and conditional entropy can be computed as follows.

$$H(S)=\frac{|S^+|}{|S|}\log_2\frac{|S^+|}{|S|}-\frac{|S^-|}{|S|}\log_2\frac{|S^-|}{|S|} \qquad \textbf{(2)}$$

$$H(S\,|\,a)=\frac{|S_{v+}^{(a)}|}{|S|}H(S_{v+}^{(a)})+\frac{|S_{v-}^{(a)}|}{|S|}H(S_{v-}^{(a)}) \qquad \textbf{(3)}$$

Definition 2: [Split Information] *Split Information* is used to measure the potential information generated by dividing *a* dataset *S* into *k* sub-ones. It is computed as

$$I(S)=-\sum_{i=1}^{k}\frac{|S_i|}{|S|}\log_2\frac{|S_i|}{|S|} \qquad \textbf{(4)}$$

In our problem setting, there are only two classes. Therefore, the split information for S that are divided according to attribute a can be computed as

$$I(S,a)=\frac{|S_{v+}^{(a)}|}{|S|}\log_2\frac{|S_{v+}^{(a)}|}{|S|}-\frac{|S_{v-}^{(a)}|}{|S|}\log_2\frac{|S_{v-}^{(a)}|}{|S|} \qquad \textbf{(5)}$$

Definition 3: [Information Gain Ratio] *Information Gain Ratio* is defined as the ratio between Information Gain and Split Information, that is *IGR*(*S*, *a*) = *G*(*S*, *a*)/*I*(*S*, *a*).

The concrete process is shown in Figure 2. The training set *D* is generated in the previous section contains ℓ different data samples $\{N_1, N_2, ..., N_\ell\}$. The Bootstrap resampling algorithm [26] is used to extract a data sample *I* times from the set *D* to obtain a sub-training set *S*. This step is repeated *q* times to obtain the *q* sub-training set $\{S_1, S_2, ..., S_q\}$, which can be used to generate *q* different decision trees to build a random forest. Among them, the steps that generate each decision tree T_i for $i = 1, ..., q$ are as shown in Algorithm 2.

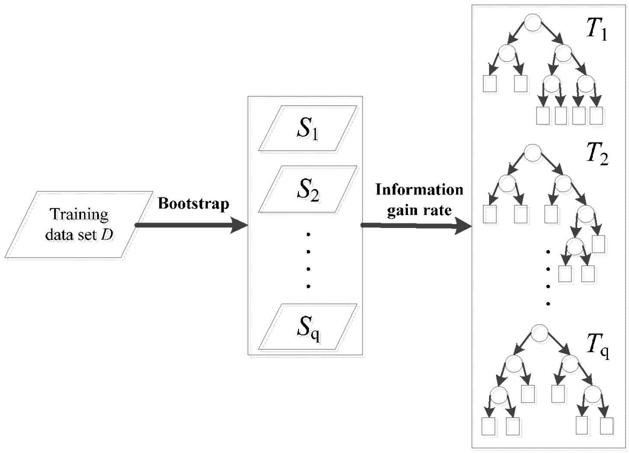


Figure 2. Traffic features extraction based on Information Gain Ratio

Algorithm 2. T_q constructing algorithm

```
Input: A feature-valued dataset $S_q$
Output: A decision tree $T_q$
1.  for each rest feature a do
2.      rank all samples according to their values such that $v_1 < \cdots < v_\ell$
3.  end for
4.  $S_q$ is divided into two sub-sets
5.  for each of the $\ell - 1$ split points do
6.      calculate the $IGR(S, a)$
7.      find the largest $IGR(S, a)$, $v_{OPT}(a)$
8.  end for
9.  select feature a with the largest IGR (S, a) as the root of current tree
10. split the current partition of datasets into two parts according to value $v_{OPT}(a)$
11.  remove $a$ from the feature set
12. repeat
13.     step 1 to 11
14. until there are no feature left or each partition is homogeneous (i.e., contain samples of only one class)
15. return $T_q$
```

The above process is illustrated in Figure 3. Firstly, a data set S_q is randomly selected from the training set by using the Bootstrap resampling algorithm. The information gain ratio of 18 features in the S_q is calculated. For each specific feature, we select the optimal split value according to step 2) in the above, and let the corresponding information gain ratio at the split value be the (best) information gain ratio for current feature. Assume that the feature with maximum information gain ratio is *Measurement* with split value at 0.397, and then it is utilized to divide data set S_q into two parts such that the information gain ratio is maximized. Assume that the samples in *Sq* is classified as abnormal while *Measurement*≤ 0.397, and the features are further extracted while *Measurement*> 0.397. When Measurement> 0.397, *Command_address* is selected as the second feature with the highest information gain ratio, which with the best split value as 0.018, in the rest features. Afterwards, the above steps are recursively performed until all features have been used or each leaf node is homogeneous, *i.e.*, contains samples of only one class.

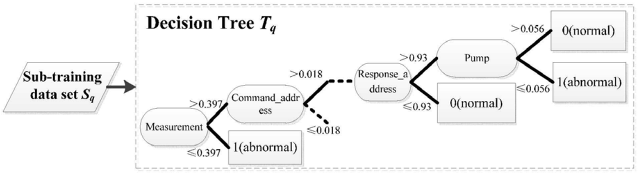


Figure 3. An example of the generation of a single decision tree

In the above method, the feature selection is carried based on information gain ratio rather than information gain. The reason is that, there is an intrinsic bias in information gain, which prefers the feature with high cardinality [27], *i.e.*, when a feature has a large number of different values (*e.g.*, *Command_memory*). The split information shown in Definition 2 for a feature refers to the entropy of the corresponding attribute expressed in the particular training set, *e.g.*, S_q. It is, in fact, a kind of normalization of information gain such that the gain ratio can be fairly compared among features with different cardinalities. When the information gain is fixed, the importance of the feature will decrease with the increase in the corresponding split information.

3.3 Attack Detection Based on Information Gain Using Weighted Majority Algorithm (WMA)

As shown in Figure 4, the Weighted Majority Algorithm (WMA) is introduced to assign an independent weight to each particular decision tree $T_{i.}$ The data gain $G(D, S_i)$, which is described by the following equation, is used to measure the weight of each decision tree T_i on the final test result.

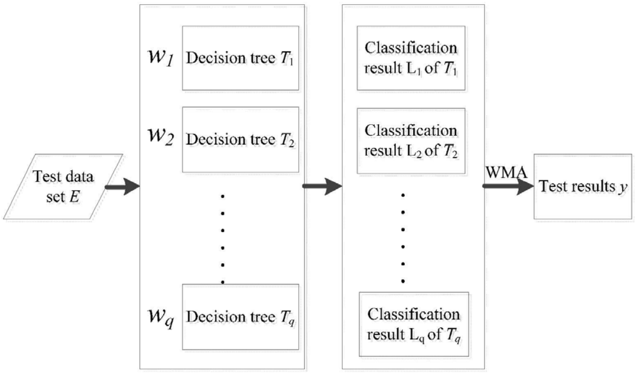

Test data set E
w_1
Decision tree T_1
w_2
Decision tree T_2
w_q
Decision tree T_q
Classification result L_1 of T_1
Classification result L_2 of T_2
Classification result L_q of T_q
WMA
Test results y

Figure 4. Attack detection based on Information Gain Ratio using WMA

$$Gain(D, S_i) = H(D) - H(S_i) \qquad \textbf{(6)}$$

For each test sample x, each decision tree will output the inferred class label for it, *i.e.*, normal or abnormal, which are denoted by 1 or −1. Then we obtain q classification results $(y_1(x), y_2(x), ..., y_q(x))$, where $y_i(x) \in \{1,-1\}$. Afterwards, the final class label for x, namely $y(x)$, can be inferred using the following equation, where *sgn* is the symbol function, *i.e.*, when $\sum_{i=1}^{q} Gain(D,S_i)\cdot y_i(x) \geq 0$, $y(x) = 1$; when $\sum_{i=1}^{q} Gain(D,S_i)\cdot y_i(x) < 0$, $y(x) = -1$.

$$y(x) = \operatorname{sgn}\left(\sum_{i=1}^{q} Gain(D,S_i)\cdot y_i(x)\right) \qquad \textbf{(7)}$$

4 Evaluation

4.1 Experiment Setup

An empirical study is conducted over three real-world datasets. *Gas* and *Water* are provided by Center of Critical Infrastructure Protection at the Mississippi State University. Both of them are released in 2014 and have been vastly adopted in IDS studies. Both contain data of industrial control systems: a gas pipeline and a water storage. The number of samples in Gas and Water are 97019 and 35774, respectively. Besides, the *NSL-KDD* dataset is also utilized. It is currently the most popular benchmark data for unknown attack detection. The 10-fold cross validation method is used to evaluate the learning performances for all the schemes, and we report the average performance over 10 runs for each experiment.

All the following experiments are performed on a PC with 3.3GHz Core 4 Duo CPU, 4GB DDR3-1600 RAM, and Microsoft Windows 7-64bit operating system. All the algorithms are implemented in Python 2.7.

4.2 Experiment Results

Feature selection performance. We compare it with a group of baselines as follows. Firstly, all the 18 common features (as shown in Figure 5) are utilized which appear in both *Gas* and *Water*; secondly, we manually select the unique features for training; thirdly, the dataset is preprocessed by using Principle Component Analysis (PCA). The features extracted using these baselines are then fed to a traditional Random Forest algorithm. We report the detection performances as well as ours within Figure 6.

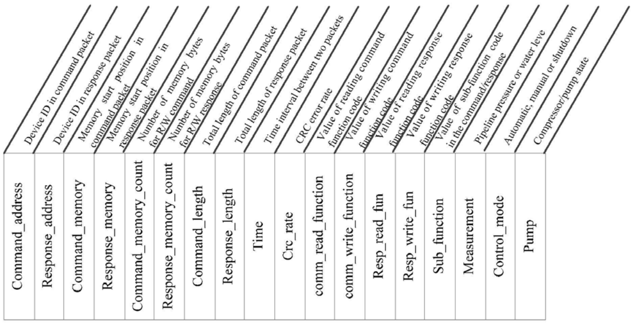
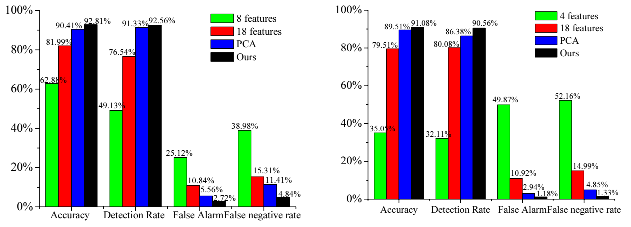

Device ID in command packet
Device ID in response packet
Memory start position in command packet
Memory start position in response packet
Number of memory bytes for R/W command
Number of memory bytes for R/W response
Total length of command packet
Total length of response packet
Time interval between two packets
CRC error rate
Value of reading command function code
Value of writing command function code
Value of reading response function code
Value of writing response function code
Value of sub-function code in the command/response
Pipeline pressure or water leve
Automatic, manual or shutdown
Compressor/pump state
Command_address
Response_address
Command_memory
Response_memory
Command_memory_count
Response_memory_count
Command_length
Response_length
Time
Crc_rate
comm_read_function
comm_write_function
Resp_read_fun
Resp_write_fun
Sub_function
Measurement
Control_mode
Pump

Figure 5. Network traffic features released in [8]

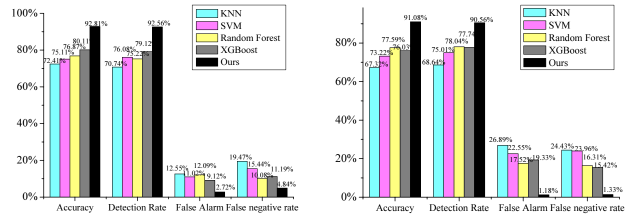


(a) Gas (b) Water

Figure 6. Performance of feature selection

As shown in Figure 6, the performance of our scheme is much better than all the other baselines. This is mainly because that different feature selection methods have lost the original information. The training model can only detect a specific attack [28-30]. These baselines may either lose a large amount of information or contain redundant or noisy data, such that they are unable to describe the traffic features. As a result, the unknown attacks cannot be effectively detected. In comparison, our scheme preserves the initial information of the dataset, so that the training model can better describe the network traffic features and have stronger generalization capability to identify different unknown attacks.

Comparison with other learning models. Secondly, this scheme is compared with the standard algorithms: Random Forest (RF), k Nearest Neighbor (kNN), Support Vector Machine (SVM) and XGBoost. The results are shown in Figure 7.



(a) Gas (b) Water

Figure 7. Comparison with other learning models

Compared with XGBoost [31], the state-of-the-art learning model that has been widely recognized in a number of machine learning challenges (*e.g.*, Kaggle and KDD Cup), our scheme is superior substantially. The *FNR*s of our scheme are lower than that of XGBoost by 56.75% and 91.37%. The results indicate that training model used in XGBoost is underfitting and cannot extract effective features for testing accurately. In comparison, we employ semi-supervised learning algorithm such that the number of labelled samples in training set are significantly enlarged. The reasons are as follows. Firstly, features in our scheme are selected based on the information gain ratio, which quantitatively finds the most discriminative features to construct the learning model. Secondly, more information have, more weights in voting.

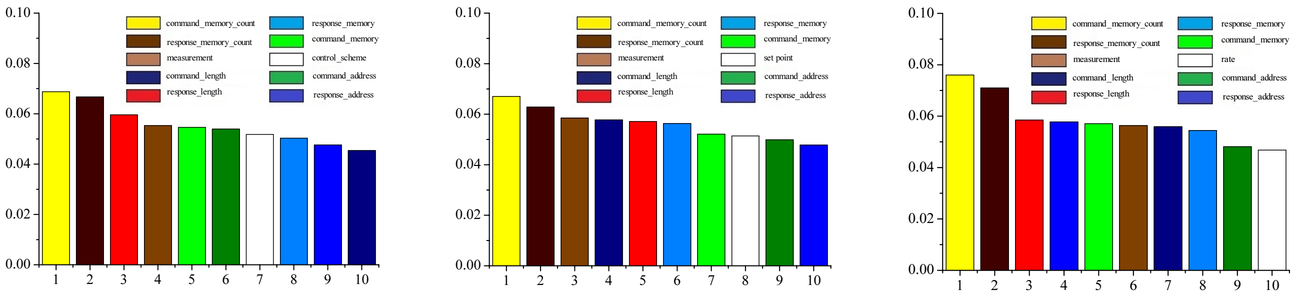
Besides, we also perform similar experiments in *NSL-KDD* dataset. The results are summarized in Table 2. Notably, our scheme shows excellent detection performance for unknown attacks as the Acc of our scheme in each dataset is higher than 90%. This justifies that our scheme has the ability to detect unknown attacks in different networks.

Table 2. The Detection performance of our scheme indifferent datasets

Data set	Acc	DR	FPR	FNR
NSL-KDD	90.48%	89.01%	2.45%	6.29%
Gas	92.81%	92.56%	2.72%	4.84%
Water	91.08%	90.56%	1.18%	1.33%

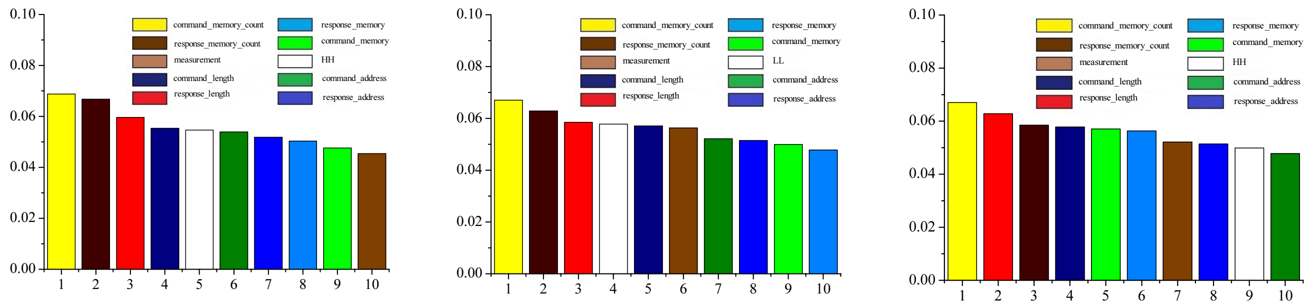
Coping with the evolution of attacks. In order to study the effect of our scheme in the light of the evolution of features in unknown attacks, another group of experiments are conducted, where *Gas* and *Water* are manually split into three subsets, respectively. Each subset contains the samples within a particular period. We then study the importance (*i.e.*, Information Gain Ratio) of all features in different time periods, respectively.

As shown in Figure 8 and Figure 9, there are two interesting phenomenon. Firstly, the features selected in different time periods are slightly different. Secondly, for the same feature, the information gain ratio in three periods is different. It justifies that, the features of the network traffic data for unknown attacks keep evolving. Due to the facts, existing works cannot deal with such change in practice. In comparison, our scheme, which quantitatively and automatically finds the most discriminative features, can successfully deal with the dynamic change of attacks.



(a) The first time period (b) The second time period (c) The third time period

Figure 8. The important features in Gas during different periods (top-10)



(a) The first time period

(b) The second time period

(c) The third time period

Figure 9. The important features in Water during different periods (top-10)

The experimental results show that this scheme can effectively generate large scale accurate labelled training sets by using a small number of labelled samples with ground-truth to ensure the effectiveness of the model training process, and can accurately extract the features of the important network traffic in the target network, and guarantee the accuracy of the model in the process of adaptive detection of different attacks.

5 Conclusion

In this paper, we propose a network traffic-based scheme for unknown attack detection based on semi-supervised learning and information gain ratio aware random forest. With the help of carefully designed an improved k-means driven semi-supervised learning algorithm, the training set can be accurately enlarged. Besides, to evaluate and select discriminative features automatically, an information gain ratio aware random forest is presented, where the final decision result is produced by WMA. As a result, we can not only quantitatively select the most discriminative features for unknown attack detection, but also improve the detection performance against the evolution of dynamic unknown attacks. Extensive experiments over three real-world datasets justify that the proposed model is both effective and efficient comparing to series of baselines.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant Nos. U1708262, U1736203, 61772173, 61672413) and National Key R&D Program of China (2017YFB0801805).

References

[1] W. Lee, S. J. Stolfo, K. W. Mok, A Data Mining Framework for Building Intrusion Detection Models, 1999 *IEEE Symposium on Security and Privacy*, Oakland, CA, 1999, pp. 120-132.

[2] M. Roesch, Snort: Lightweight Intrusion Detection for Networks, *Proceedings of the 13th Conference on Systems Administration*, Seattle, WA, 1999, pp. 229-238.

[3] B. A. Tama, K. H. Rhee, Performance Evaluation of Intrusion Detection System Using Classifier Ensembles, *International Journal of Internet Protocol Technology*, Vol. 10, No. 1, pp. 22-29, January, 2017.

[4] T. Subbulakshmi, A Learning-based Hybrid Framework for Detection and Defence of DDoS Attacks, *International Journal of Internet Protocol Technology*, Vol. 10, No. 1, pp. 51-60, January, 2017.

[5] I. V. Kotenko, A. Chechulin, D. Komashinsky, Categorisation of Web Pages for Protection against Inappropriate Content in the Internet, *International Journal of Internet Protocol Technology*, Vol. 10, No. 1, pp. 61-71, January, 2017.

[6] R. E. Maleki, M Gharib, M. Khosravi, A. Movaghar, IDS Modelling and Evaluation in WANETs against Black/Grey-hole Attacks Using Stochastic Models, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 27, No. 3, pp. 171-186, January, 2018.

[7] J.-Q. Li, Z.-F. Zhao, R.-P. Li, Machine Learning-based IDS for Software-defined 5G Network, *IET Networks*, Vol. 7, No. 2, pp. 53-60, March, 2018.

[8] J. Du, P. Xie, J.-L. Zhu, R.-J. Zheng, Q.-T. Wu, M.-C. Zhang, Method for Detecting Abnormal Behaviour of Users Based on Selective Clustering Ensemble, *IET Networks*, Vol. 7, No. 2, pp. 85-90, March, 2018.

[9] R. A. R. Ashfaq, X.-Z. Wang, J. Huang, H. Abbas, Y.-L. He, Fuzziness Based Semi-supervised Learning Approach for Intrusion Detection System, *Information Sciences*, Vol. 378, pp. 484-497, February, 2017.

[10] W. L. Al-Yaseen, Z. A. Othman, M. Z. A. Nazri, Multi-level Hybrid Support Vector Machine and Extreme Learning Machine Based on Modified k-means for Intrusion Detection System, *Expert Systems with Applications*, Vol. 67, pp. 296-303, January, 2017.

[11] E. Viegas, A. O. Santin, A. Franca R. P. Jasinski, V. A. Pedroni, L. S. Oliveira, Towards an Energy-efficient Anomaly-based Intrusion Detection Engine for Embedded Systems, *IEEE Transactions on Computers*, Vol. 66, No. 1, pp. 163-177, January, 2017.

[12] Y. Zhu, J. Liang, J. Chen, Z. Ming, An Improved Nsga-iii Algorithm for Feature Selection Used in Intrusion Detection, *Knowledge-Based Systems*, Vol. 116, pp. 74-85, January,

2017.

[13] Q.-B. Yan, Y. Zheng, T. Jiang, W.-J. Lou, Y. T Hou, Peerclean: Unveiling Peer-to-peer Botnets through Dynamic Group Behavior Analysis, *2015 IEEE Conference on Computer Communications*, Hong Kong, China, pp. 316-324, 2015.

[14] J. Li, L.-C. Sun, Q.-B. Yan, Z.-Q. Li, W. Srisaan, H. Ye, Significant Permission Identification for Machine Learning Based Android Malware Detection, *IEEE Transactions on Industrial Informatics*, Vol. 14, No. 7, pp. 3216-3225, July, 2017.

[15] X.-F. Chen, J. Li, J. Weng, J.-F. Ma, W.-J. Lou, Verifiable Computation over Large Database with Incremental Updates. *IEEE Transactions on Computers*, Vol. 65, No. 10, pp. 3184-3195, January, 2016.

[16] X.-F. Chen, J. Li, X.-Y. Huang, J.-F. Ma, W.-J. Lou, New Publicly Verifiable Databases with Efficient Updates. *IEEE Transactions on Dependable and Secure Computing*, Vol. 12, No. 5, pp. 546-556, January, 2015.

[17] X.-F. Chen, J. Li, J.-F. Ma, Q. Tang, W.-J. Lou, New Algorithms for Secure Outsourcing of Modular Exponentiations. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25, No. 9, pp. 2386-2396, January, 2014.

[18] J. Shen, T.-Q. Zhou, X.-F. Chen, J. Li, W. Susilo, Anonymous and Traceable Group Data Sharing in Cloud Computing, *IEEE Transactions on Information Forensics and Security*, Vol. 13, No. 4, pp. 912-925, January, 2018.

[19] J. Shen, T.-Q. Zhou, D.-B. He, Y.-X Zhang, X.-M. Sun, Y. Xiang, Block Design-based Key Agreement for Group Data Sharing in Cloud Computing, *IEEE Transactions on Dependable and Secure Computing*, Vol. 99, July, 2017.

[20] J. Shen, J. Shen, X.-F. Chen, X.-Y. Huang, W. Susilo, An Efficient Public Auditing Protocol with Novel Dynamic Structure for Cloud Data, *IEEE Transactions on Information Forensics and Security*, Vol. 12, No. 10, pp. 2402-2415, January, 2017.

[21] J. Shen, T.-Q Zhou, X.-G. Liu, Y.-C. Chang, A Novel Latin Square-based Secret Sharing for M2M Communications, *IEEE Transactions on Industrial Informatics*, Vol. 14, No. 8, pp. 3659-3668, February, 2018.

[22] K. Jaswal, P. Kumar and S. Rawat, Design and Development of a Prototype Application for Intrusion Detection Using Data Mining, *4th International Conference on Reliability, Infocom Technologies and Optimization*, Noida, India, 2015, pp. 1-6,

[23] O. Chapelle, B. Scholkopf, A. Zien, Semi-supervised learning, *IEEE Transactions on Neural Networks*, Vol. 20, No. 3, pp. 542-542, January, 2009.

[24] S. Basu, M. Bilenko, R. J. Mooney, A Probabilistic Framework for Semi-supervised Clustering, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, WA, 2004, pp. 59-68.

[25] L. Breiman, Random Forests, *Machine Learning*, Vol. 45, No. 1, pp. 5-32, October, 2001.

[26] J. Rice, Mathematical Statistics and Data Analysis, *Nelson Education*, Vol. 31, pp. 390-391, January, 1988.

[27] S. L. Salzberg, C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993, *Machine Learning*, Vol. 16, No. 3, pp. 235-240, September, 1994.

[28] G. Wang, J.-X. Hao, J. Ma, L.-H. Huang, A New Approach to Intrusion Detection Using Artificial Neural Networks and Fuzzy Clustering, *Expert Systems with Applications*, Vol. 37, No. 9, pp. 6225-6232, January, 2010.

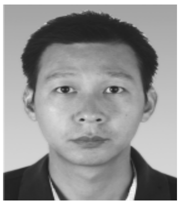
[29] X. Zhang, C. Gu, G. Lin, Intrusion Detection System Based on Feature Selection and Support Vector Machine, *International Conference on Communications and Networking*, Beijing, China, 2006, pp. 1-5.

[30] G. Liu, Z. Yi, S. Yang, A Hierarchical Intrusion Detection Model Based on the PCA Neural Networks, *Neurocomputing*, Vol. 70, No. 7, pp. 1561-1568, December, 2007.

[31] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, 2016, pp. 785-794.

Biographies

Meng-Fan Xu, born in 1989. Ph.D. candidate. His main research interests include network and information security, APT attack detection.



Xing-Hua Li, born in 1978. Professor and Ph.D. supervisor in Xidian University. His main research interests include wireless networks security, privacy protection, cloud computing, software defined network and security protocol formal methodology.

Mei-Xia Miao, born in 1980. Ph.D. candidate. The main research areas are cloud computing security and data security.

Cheng Zhong, born in 1994. M.S.c candidate. His main research interests include network and information security, intrusion detection.

Jian-Feng Ma, born in 1963. Professor and Ph.D. supervisor in Xidian University. Member of CCF. His main research interests include information security, coding theory, and cryptography.