Computers, Materials & Continua CMC, vol.65, no.2, pp.1321-1334, 2020

Ensemble Strategy for Insider Threat Detection from User Activity Logs

Shihong Zou[1], Huizhong Sun[1,*], Guosheng Xu[1] and Ruijie Quan[2]

Abstract: In the information era, the core business and confidential information of enterprises/organizations is stored in information systems. However, certain malicious inside network users exist hidden inside the organization; these users intentionally or unintentionally misuse the privileges of the organization to obtain sensitive information from the company. The existing approaches on insider threat detection mostly focus on monitoring, detecting, and preventing any malicious behavior generated by users within an organization's system while ignoring the imbalanced ground-truth insider threat data impact on security. To this end, to be able to detect insider threats more effectively, a data processing tool was developed to process the detected user activity to generate information-use events, and formulated a Data Adjustment (DA) strategy to adjust the weight of the minority and majority samples. Then, an efficient ensemble strategy was utilized, which applied the extreme gradient boosting (XGBoost) model combined with the DA strategy to detect anomalous behavior. The CERT dataset was used for an insider threat to evaluate our approach, which was a real-world dataset with artificially injected insider threat events. The results demonstrated that the proposed approach can effectively detect insider threats, with an accuracy rate of 99.51% and an average recall rate of 98.16%. Compared with other classifiers, the detection performance is improved by 8.76%.

Keywords: Insider threat, data adjustment, imbalanced data, ensemble strategy.

1 Introduction

Nowadays, with the rapid development of networks, many enterprises/organizations have set up their own inside network. Although the inside network has facilitated the work and the management of the organization, the network security incidents are happening more and more frequently and becoming more serious. Recent reports have shown that 90% of the organizations feel vulnerable to insider attacks. The main enabling risk factors include too many users with excessive access privileges (37%), the increasing number of devices with sensitive data access rights (36%), and the increasing complexity of information technology (35%) [Crowd Research Partners (2019)].

Insider threat-related activities mainly come from within the organization and pose a threat to the organization itself, such as involving current or former employees,

[1] School of Cyberspace Security, Beijing University of Posts and Telecommunication, Beijing, China.

[2] Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia.

[*] Corresponding Author: Huizhong Sun. Email: sunhuizhong@bupt.edu.cn.

Received: 14 January 2020; Accepted: 01 June 2020.

contractors, or partners who have or have authorized access to the organization's networks, systems, or data and abuse the access rights. But in most cases, users ignore the form of "unintentional" insider threats. For example, when a user accidentally sends an email containing sensitive information to the wrong recipients. Although users do not intend to conduct potential threats, their actions may indeed have serious consequences for the company, and may even cause the same damage as "intentional" insider threats. Similar insider threats with "malicious intentions" are usually submerged in massive amounts of normal data, and any proposed system for insider threat detection needs to overcome the challenges in learning from the highly imbalanced data of heterogeneous sources in order to distinguish malicious activities from the legitimate ones, all of which are from authorized users.

Therefore, to be able to detect insider threats more effectively, a data processing tool was developed to process the detected user activity to generate information-use events and formulated DA strategy to adjust the weight of the minority and majority samples. Then, an efficient ensemble strategy was utilized, which applied the XGBoost [Chen, He, Benesty et al. (2015)] model combined with the DA strategy to detect an insider threat. The insider threat dataset (see Tab. 1) was used to evaluate the proposed approach, which contains multiple types of insider threat scenarios and is specifically used for insider threat detection system evaluation. The results showed that the proposed ensemble strategy was effective in insider threat detection; this is of practical significance to the detection of insider threats in organizations.

2 Related work

In the field of internal threat detection, the current research on this issue has attracted the attention of many government organizations and security companies. Many researchers have proposed various solutions to detect attacks from inside network users within organizations. Hunker et al. have presented an overview of the definition of insider threats and discussed some approaches from the domains of technology, sociology, and social technology [Hunker and Probst (2011)]. Their main conclusion was that dealing with insider threats requires combining technologies from these domains to detect and mitigate insider threats. Many insider threat detection systems are derived from DARP's ADAMS project [Thompson, Stolfo, Keromytis et al. (2011); Le and Zincir-Heywood (2018); Eldardiry, Bart, Liu et al. (2013); Rashid, Agrafiotis, Nurse et al. (2016); Gavai, Sricharan, Gunning et al. (2015); Goldberg, Young, Reardon et al. (2017)], which aimed to identify patterns and anomalies in large datasets to address insider threats. Le et al. [Le and Zincir-Heywood (2018)] proposed an unsupervised self-organizing graph learning algorithm for distinguishing normal user and malicious user activities. Eldardiry et al. [Eldardiry, Bart, Liu et al. (2013)] proposed an approach to use a hybrid anomaly detector to detect anomalous changes from information combined from multiple domains (user activity). Rashid et al. [Rashid, Agrafiotis, Nurse et al. (2016)] applied the hidden Markov model (HMM) to model a user's weekly activity sequence and detected possible insider threats from subtle changes. Gavai et al. [Gavai, Sricharan, Gunning et al. (2015)] used different approaches based on machine learning to construct user activity data under specific threat scenarios for insider threat detection. Goldberg et al. [Goldberg, Young, Reardon et al. (2017)] proposed a combination of Gaussian algorithm and hidden Markov

model to organize user activity log data to identify insiderl threat indicators.

An online unsupervised deep learning approach is proposed to detect anomalous network activity in the system logs in real time [Tuor, Kaplan, Hutchinson et al. (2017)]. The model decomposed the anomaly score into the contribution of individual user behavior features to improve the security analysts' review of potential insider threat activities. In order to detect insider threats from complex audit data, Liu et al. [Liu, DeVel, Chen et al. (2018)] proposed an anomaly detection system based on a deep autoencoder. Each autoencoder was trained using a specific category of audit data that accurately represented the employee's normal behavior. Azaria et al. [Azaria, Richardson, Kraus et al. (2014)] presented a framework for insider threat (BAIT) behavior analysis, and they conducted detailed experiments on 795 subjects on Amazon Mechanical Turk (AMT) to assess real human subjects when attempting to exfiltrate data from within the organization. In the real world, the number of actual insiders found is very small [Luo, Wang, Cai et al. (2019)]; therefore, the approach of machine learning has encountered challenges.

It can be clearly observed from these related works that the insider threat of the enterprise/organization is the result of a range of abnormal activities caused by the inside network users with malicious intentions abusing the organization's resources. As a result, a small number of malicious activities are embedded in a large amount of normal activity data. How to detect malicious activity behavior from imbalanced ground-truth insider threat data is a significant challenge in current machine learning. Therefore, we propose an ensemble strategy for insider threat detection from user activity logs to address this problem. The main contributions are as follows:

(1) In this study, a data processing tool was developed to address the activity log data to represent user behavior and formulated a DA strategy to optimize imbalanced data.

(2) Combining the advantages of the XGBoost model and the DA strategy, this paper constructed an ensemble strategy to detect insider threats.

(3) The results demonstrated that the proposed approach can effectively detect insider threats, with an accuracy rate of 99.51% and an average recall rate of 98.16%. Compared with other classifiers, the detection performance is improved by 8.76%.

3 System overview

An overview of the proposed novel ensemble strategy for insider threat detection from user activity logs is shown in Fig. 1. The log data collection, data processing, and feature extraction steps of the inside network users of the organization are detailed in Section 3.2. The next step is to use DA strategies to optimize imbalanced data as described in Section 3.4.

In Section 3.5, we present an ensemble strategy, which combines the strengths of the XGBoost model and the DA strategy to detect insider threats. In the experimental analysis section, (i) We use the CERT data set as test data, and apply a data processing tool to extract feature, such as statistical user email records, access times to log in to the office system, use times of external devices, etc. (ii) We construct different insider threat scenarios and organize user features as data analysis. The evaluation metrics we use are precision, recall, F1-score, and ROC curve to verify the performance of the proposed approach.

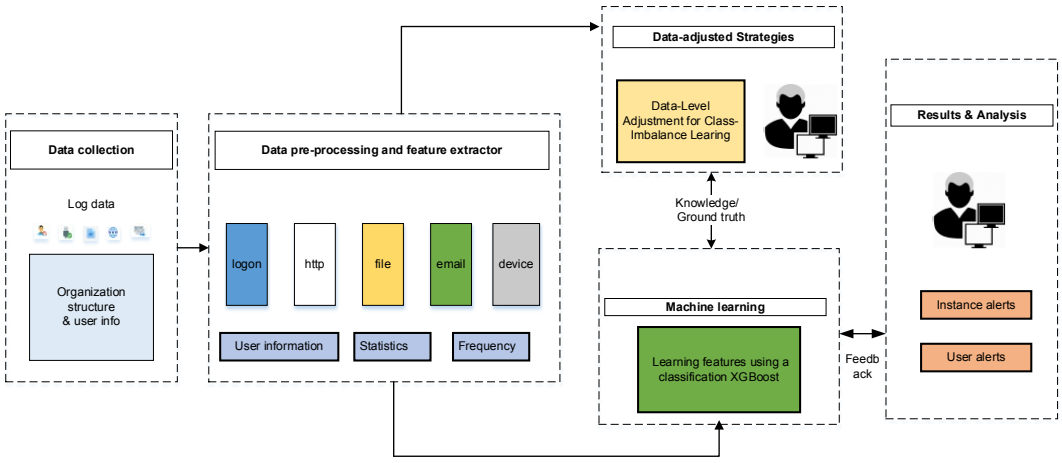

Figure 1: Insider threat detection model

3.1 Data collection, processing and feature extraction

On the basis of the daily activities of the inside network users of the organization, we analyzed the behavior of inside network users. Therefore, it is necessary to collect behavior information of the users. It was specifically collected log files include web server access logs, email server access logs and file server access logs in the organization inside network. we use these data is to correlate the information recorded by such activity log data to directly or indirectly reflect the user's regular pattern and interest. An effective monitoring process combined with adequate data collection can successfully apply machine learning techniques and enable security analysts to make the correct decisions.

Therefore, in this study, we employ the CERT insider threat dataset to validate our approach, which is synthesized from the organization's individual server log data. It includes login events, device events, mail events, Http events, file events, etc. we have processed the dataset to extract data features and provide suitable data formats for machine learning algorithms. But in many cases, various types of user data do not provide enough features to characterize complete user information. therefore, we need to reason about the user's related activity characteristics based on existing data and use it as auxiliary information for further processing. In particular, we design and describe the user's activity relationship according to the sequence of user activities, such as the user's activity relationship in the same department, the number of times the external device is used during and outside working hours, the relationship between users accessing internal files during normal working hours and non-working hours, working time access website category, etc. For instance, in the logon event activity, the relationship <userid, PC, user name, date, logon time, and logoff time> was extracted from the log file. In the device event activity, the relationship <userid, PC, user name, date, connect, and disconnect> was extracted from the log file. In the email event activity, the relationship <userid, PC, user name, date, the number of external mails sent per day, the number of inside emails sent every day, the number of email attachments sent every day> was extracted from the device email file. In the file event activity, the relationship <userid, PC, user name, date, activity, sensitive file access frequency, download file size, upload file size> was

extracted from the log file. In the http event activity, the relationship <userid, PC, username, date, access wikileaks.org frequency, key-logger downloading sites frequency, job-advertisement related web-pages frequency> was extracted from the log file. After obtaining the user attribute-related information, we constructed the daily activity information on the basis of the time sequence for each user.

Table 1: Single-Day features

Log Type	List of Single-Day Features
Logon	difference of office start time from first login time
	difference of last login time from office end time
	average time difference between login times in office hours
	average time difference between login times after office hours
	total number of logins
	total number of logouts
	number of logins outside office hours
	total number of computers accessed
	number of computers accessed outside office hours
	average session time outside office hours
Device	number of times a thumb drive was used outside office hours
	number of times a thumb device was used during non-office hours
	total number of times a device was used
Http	total number of times wikileaks.org was visited
	TF-IDF-based feature for job-advertisement-related webpages
	TF-IDF-based feature for key-logger downloading sites
File	number of times decoy files were copied
	number of times .exe files were downloaded
Email	number of emails sent outside the organization's domain
	number of emails sent inside the organization's domain from the supervisor's account
	number of attachments
	average email size
	number of recipients

Based on the data information obtained in the previous step, feature extraction is shown in Tab. 1. In this work, we extract two types of features that characterize the user's complete information from various types of user activity data: (i) statistical features, especially the mean and standard deviation of the data, e.g., internal email attachment size, external email attachment size, sensitive files size, or the number of keywords in the recruitment website visited, and (ii) frequency features, counting the number of user activities within a certain period of time, e.g., the number of user logins and logouts, the number of external emails sent, the number of internal emails, the number of accesses to sensitive files, or the number of external device insertions. Use these two types of feature data to completely describe

user information to provide the context of machine learning algorithms.

3.2 Data-adjusted strategies

In this section, we analyze the outcomes of the pre-processed feature data. The distribution of features is shown in Fig. 2. In the class, 0 represents a sample of malicious features, and 1 represents a sample of normal features. This figure shows that the ratio of malicious samples to normal samples is extremely imbalanced. Machine learning algorithms learn from the imbalanced data, and training and detection lead to a high false alarm rate, that is, ignore the real malicious behavior. To this end, we propose a DA strategy to optimize imbalanced data.

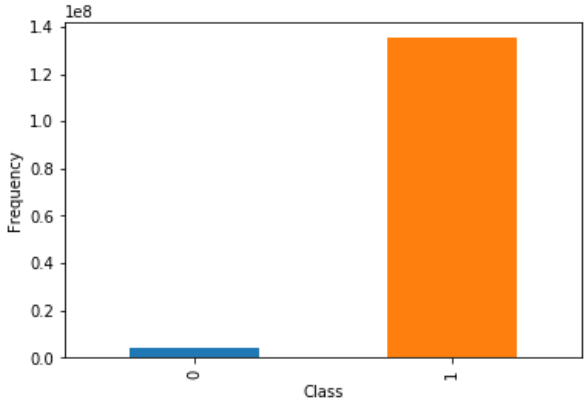

Figure 2: Histogram of normal and malicious sample distributions

Thus far, many researchers have achieved good efforts in dealing with imbalanced data classification issues [Galar, Fernandez, Barrenechea et al. (2011)], such as resampling strategies: undersampling, oversampling, and hybrid approaches. The synthetic minority oversampling technique (SMOTE) is a synthetic sub-sampling technique. It is an improved scheme based on a random oversampling algorithm. However, as random oversampling adopts a strategy of simply copying samples to increase the small number of samples, it is easy to produce an issue of model overfitting. The solution to Borderline-SMOTE, is to add small samples of the k-nearest neighbor boundaries [Qu, Li, Xu et al. (2019)]. In addition, random undersampling (RUS) is a very simple integrated algorithm for imbalanced data sets, which has attracted the interest of many researchers with its efficient calculation method. In this paper, we apply a DA strategy, which combines SMOTE (for a minority), RUS (for a majority), and XGBoost, which combines the advantages of resampling methods and classifiers. Firstly, we present the XGBoost algorithm to obtain the malicious feature (MS) sample. Secondly, the RUS algorithm is used to reduce the proportion of most types of data samples. Finally, the SMOTE algorithm increases the proportion of minority data samples. We extracted two types of instances, such as misclassified instances belonging to minority classes (MS+) and misclassified instances belonging to majority classes (MS-), The detailed steps are presented in Algorithm 1:

Algorithm 1 Data-Adjusted Strategies

Input: Training set S = $\{x_i,y_i\}$,i =1,2, ..., N+M;Label y_i = {0,1},where y_i = 0 represents a minority class and y_i = 1 represents a majority class.

Output: Optimized dataset.

1: Let S=$\{x_i,y_i\}$ divides into majority set MX = $\{x_i,y_i\}$,i=1,2,..., M and minority set MN=$\{x_i,y_i\}$, i=1, 2, ..., N;

2:IR (imbalance ratios) = N/M;

3: **if** IR≥500 **then**

4: n=500

5: **else**

6: n=IR

7: **end if**

8: Let MX=$\{x_i,y_i\}$ divided into n non-overlapping subsets;

9: Each MX subset combined with MN = $\{x_i,y_i\}$ to get n data subsets

Sub = $\{Sub_i\}$, i=1,2, ..., n, where Sub_i =$\{ x_i,y_i \}$,i=1,2, ..., (N+M/n);

10: Initialize, MS += {} and MS −= {} is null set;

11: i**f** (N+M)<3000 and IR>100 **then**

12: Using XGBoost algorithm to extract MS from training samples;

13: **else**

14: **for** each i=[2, n] **do**

15: Generate the 10-flod cross validation set Val=$\{Val_j\}$, j=1,2, ..., k from Sub_i

16: f**or each** j=[1, 10] **do**

17: Use XGBoost algorithm to train the remaining nine sets, get $MS_{i,j}$ from Val_j by XGBoost;

18: **end for**

19: Get MS += $\{MS_{i,j}+\}$,i=1,2...n;j = 1,2, …,10 and MS −=$\{MS_{i,j}-\}$,i=1,2, ..., n;j=1,2,…,10;

20: New minority set NMN={MS+,MI} ,NMX={MS-,aMX},a∈(0,1) ;

21: Combine dataset S={NMN,NMX} based on NMN and NMX

22: According to the imbalance ratio of the S set, the Smote algorithm is used to Increase minority abnormal behavior feature data

23: **end for**

24: **end if**

It is an effective learning approach to tackle the issue of imbalanced data in datasets. Compared with the SMOTEBoost strategy [Yu, Mu, Sun et al. (2015); Qu, Wu, Wang et al. (2017)], (i) SMOTEboost combines SMOTE and boosting algorithms, uses SMOTE to increase the prediction performance of a few samples, and uses boosting to improve the overall accuracy. The proposed DA strategy is an independent non-iterative process. ((ii) SMOTEBoost only uses the update strategy for the misclassification of a few samples, and the proposed DA strategy mainly focuses on the misclassification of a few samples and the misclassification of a majority of samples.

3.3 XGBoost model

The XGBoosting tree is a boosting method based on a classification tree. It is considered to be one of the best methods in statistical learning. The linear combination of multiple trees can well fit the training data and describe the complex nonlinear relationship between the input and the output data. First, for *n* samples and *m* features, set a data set $D = \{(x_i, y_i)\}(x_i \in R^m, y_i \in (0,1), i = 1,2,...,n)$, Using K addition functions to predict output in XGBoost model:

$$\hat{y}_i = \varphi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{1}$$

where $F = \{f(x) = \omega_{q(x)}\}(q : R^m \to T, \omega \in R^m)$ is the space containing the regression tree,, $\hat{y}_i$ is the prediction, *T* is the number of leaves in the tree and y_i is the real label and. Here *w* is the weight vector of each leaf node and *q* represents the structure of a single tree. The objective function is given as following:

$$L(\varphi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{2}$$

where l represents the convex loss function and $\Omega(f) = \gamma T + \lambda \|\omega\|^2$ represents an addition model, representing the penalty function. We use the following formula to obtain a value for evaluating the performance of a given *q*:

$$\hat{L}^{(t)}(q) = -\frac{1}{2}\sum_{j=1}^{T} \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \tag{3}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$ and $h_i = \partial^2_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{-(t-1)})$ are the gradient and the second-order gradient functions. The smaller the score, the better the structure *q*.

4 Experiments and results

In this section, we evaluate the proposed insider threat detection approach and verify the performance metrics on the CERT r6.2 dataset [Insider Threat Program Development (2017)]. As discussed in Section 4.1, we used the CERT insider threat dataset, which is a publicly available dataset for the research, development, and testing of insider threat mitigation approaches. In Section 4.2, we describe the experimental setup and parameter optimization. In Section 4.3, we discuss the results and the analysis of the performance anomaly detection approach.

4.1 Dataset employed

The CERT r6.2 dataset consists of event log lines from a simulated organization's computer network, generated with sophisticated user models. We used five sources of events: logon/logoff activity, http traffic, email traffic, file operations, and external storage device usage. Over the course of 516 days, 4,000 users generated 135,117,169 events (log lines). Among these were the events manually injected by the domain experts, representing

five insider threat scenarios taking place. For details of these insiderl threat scenarios, please refer to the method proposed by Glasser et al. [Glasser and Lindauer (2013)].

On the basis of the original CERT data, we performed the data processing steps to obtain the single-day feature data, as described in Section III-B. According to the feature count, the pre-processed single-day data had 1092 features. Fig. 2 shows the distribution of the CERT r6.2 data according to the numbers of normal and malicious users. Obviously, the distribution of data samples is extremely uneven, and the malicious data of rare sample data only accounts for 0.12% of the single-day data. Additionally, user attribute metadata were included. Tab. 2 shows the dataset user activity log attribute statistics.

Table 2: Summary of the dataset

Date Range	Normal	Malicious
Threat Events	135,117,169	428
Threat single-day	516	47
Users	3,995	5

4.2 Experimental setup and parameter optimization

To be able to assess the performance of the detection system, we divide the entire dataset chronologically into two subsets: training and testing. The training subset (~80% of the data) is used for large amounts of data for learning and training, so that we can choose different models and adjust the optimal parameters, while the latter subset (~20% of the data) is used to verify the test indicators, which is convenient for evaluating the generalization performance. Tab. 3 shows the XGBoost model parameters. Our predictions were made at the granularity of the user single-day data. according to the count of user events on weekdays, we found that the number of days threatening users is far less than the original events, and the threat events generated by malicious users are usually concentrated in one day or several days. Therefore, we filtered the irrelevant data of the relevant days and set the test set to contain 20% of the events, which has more than 40% of the threat user days.

Table 3: Parameters of XGBoost Model

Parameter Name	Parameter Value
Learning_rate	0.02
Max_depth	4
Subsample	0.8
Colsample_bytree	1
Lambda	10
Gamma	20
Min_child_weight	5
Colsample_bylevel	1

4.3 Results and analysis

In this section, we select the experimental data, which contain a large number of activity logs of the daily work of inside network users in the enterprise/organization. These activity data imply a variety of common and some unusual abnormal behaviors. The purpose of this experiment was to apply the insider threat detection model mentioned in Section 3 to identify malicious abnormal behaviors from these activity log data and, based on the experimental results, to evaluate the detection effect and detection efficiency. Tab. 3 shows the accuracy, recall, and F1-score comparison analysis of the DA strategy combined with other baseline classifiers (random forest (RF) and gradient boost tree (GBT)). The results proved that the proposed approach was effective in insider threat detection with a 99.51% precision at 98.16% recall on average. Compared with other classifiers, the detection performance is improved by 8.76%.

Table 4: Results of DA-based strategy combined with other baseline classifiers (random forest (RF) and gradient boost tree (GBT))

Classifier	Precision	Recall	F1-score	AUC
RF	96.78%	78.52%	86.69%	85.81%
RF+DA	97.84%	86.26%	91.68%	89.16%
GBT	95.89%	74.81%	84.04%	84.08%
GBT+DA	98.21%	87.75%	92.68%	90.35%
XGBoost	98.57%	82.92%	90.07%	91.78%
XGBoost+DA	99.51%	98.16%	98.83%	96.87%

Fig. 3 shows the significant difference between adding DA-based sampling strategy and not adding DA-based sampling strategy. The classifier with DA-based sampling strategy can be improved by 1%-8% in precision, recall, F1-score. Meanwhile, DA-based sampling showed that XGBoost performance reached 8% at F1-score. According to the analysis of experimental results, the XGBoost model combined with DA-based sampling strategy shows its outstanding advantages in solving the binary classification imbalance problem compared with other classifiers. It not only shows good performance on small imbalanced data sets without any preprocessing, but also outstanding on large-scale imbalanced data sets.

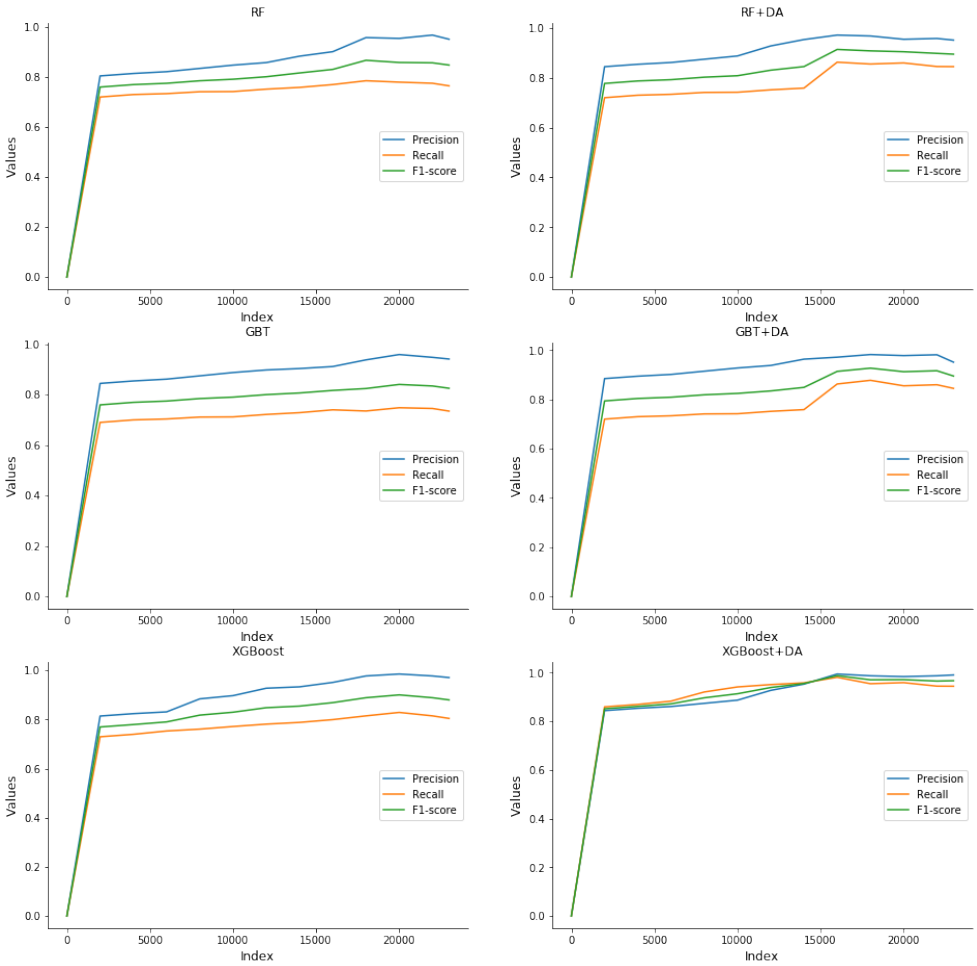

Figure 3: Metric values of baseline classifiers with DA strategies and baseline classifiers

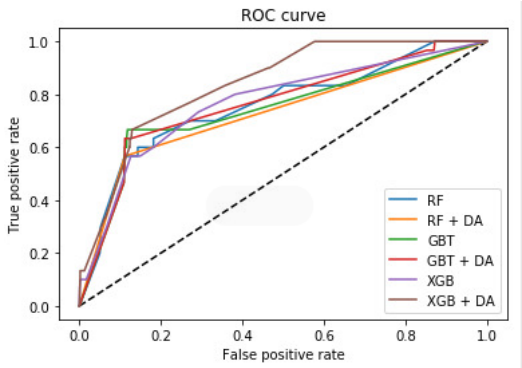

Figure 4: ROC curve from DA-based strategy combined with other baseline classifiers (random forest (RF) and gradient boost tree (GBT)

Fig. 4 shows that the ROC curve performance of the applied DA-based sampling strategy was obviously better than that of the DA strategy combined with other classifiers. the AUC score was 96.87%. For the CMU Insider Threat dataset, based on different insider threat scenarios, we could dynamically adjust the parameters to achieve the best prediction effect.

To summarize, the proposed ensemble strategy could not only improve the accuracy of insider threat prediction, but also reduces the number of false positives to reduce the workload of security managers.

5 Conclusion and future work

In this paper, we have proposed an ensemble strategy for insider threat detection from user activity logs. First, we establish a behavior analysis model and study the behavior features for insider network users. The features we develop include daily activity logs such as user login, email sending, file activity, http access and device usage. We then used these features to initial anomaly detection using a *K*-mean clustering approach: (i) we used these features as input to an unsupervised anomaly detection method in order to detect suspicious behavior and (ii) we used these features in conjunction with marking labels to develop a classifier using supervised classification methods. Then we propose a DA strategy to optimize these feature datas. Finally, we have presented a novel ensemble strategy which combines DA strategy with XGBoost model for insider threat detection on synthetic datasets.

In the future work, our definition and modeling of abnormal behavior needs further improvement. Here is mainly to consider more ehavioral factors in the model, associated with more data for abnormal behavior detection. In initial anomaly detection, how to set up dynamic cluster parameter problem in *k*-mean. In the future, we try to consider to apply the big data platform performs real-time analysis of large-scale data.

Acknowledgement: We thank LetPub (www.letpub.com) for its linguistic assistance during the preparation of this manuscript.

Funding Statement: This work was financially supported by "the National Key R&D Program of China" (No. 2018YFB0803602) and exploration and practice on the education mode for engineering students based on technology, literature and art interdisciplinary integration with the Internet+ background (No. 022150118004/001)

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Azaria, A.; Richardson, A.; Kraus, S.; Subrahmanian, V. S. (2014): Behavioral analysis of insider threat: A survey and bootstrapped prediction in imbalanced data. *IEEE Transactions on Computational Social Systems*, vol. 1, no. 2, pp. 135-155.

Chen, T.; He, T.; Benesty, M.; Khotilovich, V.; Tang, Y. (2015): XGBoost: extreme gradient boosting. *R Package Version 0.4-2*, pp. 1-4.

Crowd Research Partners (2019): "2019 insider threat report".
http://www.securonix.com/resources/2019-insider-threat-survey-report/.

Eldardiry, H.; Bart, E.; Liu, J.; Hanley, J.; Price, B. et al. (2013): Multi-domain information fusion for insider threat detection. *IEEE Security and Privacy Workshops*, pp. 45-51.

Galar, M.; Fernandez, A.; Barrenechea, E.; Bustince, H.; Herrera, F. (2011): A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463-484.

Gavai, G.; Sricharan, K.; Gunning, D.; Hanley, J.; Singhal, M. et al. (2015): Supervised and unsupervised methods to detect insider threat from enterprise social and online activity data. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 6, no. 4, pp. 47-63.

Glasser, J.; Lindauer, B. (2013): Bridging the gap: A pragmatic approach to generating insider threat data. *IEEE Security and Privacy Workshops*, pp. 98-104.

Goldberg, H.; Young, W.; Reardon, M.; Phillips, B. (2017): Insider threat detection in prodigal. *Hawaii International Conference on System Sciences.*

Hunker, J.; Probst, C. W. (2011): Insiders and insider threats: an overview of definitions and mitigation techniques. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 2, no. 1, pp. 4-27.

Insider Threat Program Development (2017): https://www.sei.cmu.edu/about/divisions/cert/.

Le, D. C.; Zincir-Heywood, A. N. (2018): Evaluating insider threat detection workflow using supervised and unsupervised learning. *IEEE Security and Privacy Workshops*, pp. 270-275.

Lemaître, G.; Nogueira, F.; Aridas, C. K. (2017): Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, vol. 18, no. 1, pp. 559-563.

Liu, L.; De Vel, O.; Chen, C.; Zhang, J.; Xiang, Y. (2018): Anomaly-based insider threat detection using deep autoencoders. *IEEE International Conference on Data Mining Workshops*, pp. 39-48.

Luo, M.; Wang, K.; Cai, Z.; Liu, A.; Li, Y. et al. (2019): Using imbalanced triangle synthetic data for machine learning anomaly detection, *Computers, Materials & Continua*, vol. 58, no. 1, pp. 15-26.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B. et al. (2011): Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, vol. 12, no. 10, pp. 2825-2830.

Qu, Z.; Li, Z.; Xu, G.; Wu, S.; Wang, X. (2019): Quantum image steganography protocol based on quantum image expansion and Grover search algorithm. *IEEE Access*, vol. 7, pp. 50849-50857.

Qu, Z.; Wu, S.; Wang, M.; Sun, L.; Wang, M. (2017): Effect of quantum noise on deterministic remote state preparation of an arbitrary two-particle state via various quantum entangled channels. *Quantum Information Processing*, vol. 16, no. 306, pp. 1-25.

Rashid, T.; Agrafiotis, I.; Nurse, J. R. (2016): A new take on detecting insider threats: exploring the use of hidden Markov models. *Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats*, pp. 47-56.

Thompson, H.; Stolfo, S. J.; Keromytis, A. D.; Hershkop, S. (2011): *Anomaly Detection at Multiple Scales (ADAMS) (No. CLIN 0001AA)*. Allure Security Technology Inc. New York NY.

Tuor, A.; Kaplan, S.; Hutchinson, B.; Nichols, N.; Robinson, S. (2017): Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. *Workshops at the Thirty-First AAAI Conference on Artificial Intelligence*.

Yu, H.; Mu, C.; Sun, C.; Yang, W.; Yang, X. et al. (2015): Support vector machine-based optimized decision threshold adjustment strategy for classifying imbalanced data. *Knowledge-Based Systems*, vol. 76, no. 1, pp. 67-78.