ISSN: 2366-1313

DATA CURATION AND QUALITY ASSURANCE FOR MACHINE LEARNING-BASED CYBER INTRUSION DETECTION

[1]D.Srivalli, [2]Sarala Lakshmi, [3]K.Snehalatha Reddy, [4]S.Cheritha, [5]Sri M.Manogna

[1]Assistant Professor, Department of CSE(DS), Malla Reddy Engineering College for Women (Autonomous Institution – UGC, Govt. of India), Hyderabad, INDIA.

[2345]UG,Department of CSE(DS), Malla Reddy Engineering College for Women (Autonomous Institution – UGC, Govt. of India), Hyderabad , INDIA.

Abstract:

Intrusion detection is an essential task in the cyber threat environment. Machine learning and deep learning techniques have been applied for intrusion detection. However, most of the existing research focuses on the model work but ignores the fact that poor data quality has a direct impact on the performance of a machine learning system. More attention should be paid to the data work when building a machine learning-based intrusion detection system. This article first summarizes existing machine learning-based intrusion detection systems and the datasets used for building these systems. Then the data preparation workflow and quality requirements for intrusion detection are discussed. To figure out how data and models affect the machine learning performance, we conducted experiments on 11 HIDS datasets using seven machine learning models and three deep learning models. The experimental results show that BERT and GPT were the best algorithms for HIDS on all of the datasets. However, the performance on different datasets vary, indicating the differences between the data quality of these datasets. We then evaluate the data quality of the 11 datasets based on quality dimensions proposed in this paper to determine the best characteristics that a HIDS dataset should possess in order to yield the best possible result. This research initiates a data quality perspective for researchers and practitioners to improve the performance of machine learning-based intrusion detection.

Keywords Data curation · Data quality · Machine learning · Intrusion detection · Host-based intrusion detection systems

1 INTRODUCTION

The increasing usage of digital devices in a cyber-physical system (CPS) has enhanced the efficiency of operating systems, but has also led to vulnerability from cyber-attacks. Cyber assaults on process control and the monitoring of

ISSN: 2366-1313

these intelligent systems could lead to a significant control failure [1] and huge economic losses. This makes cyber-security a major concern due to the high level of attacks on networks and systems for CPS [2]. Therefore, building an intrusion detection system (IDS) to predict and respond to assaults on a CPS, has become an essential task among the software engineering community. However, it is very challenging because of the range of novelty involved in cyber-attacks [1]. Recently, machine learning and deep learning have been applied for intrusion detection at both the operation system level (a host-based intrusion detection system, called HIDS) and the network level (a network-based intrusion detection system, called NIDS). As indicated by Google Research [3], both the models/algorithms and the quality of the data greatly impact the performance of machine learning systems. The computing rule of "garbage in, garbage out" is still applicable to machine learning [4] and the lack of high quality training data becomes a barrier for building high performance machine learning systems [5]. Therefore, we should not only optimize the models but also systematically evaluate and ensure the data quality to improve performance for intrusion detection.

However, current studies on machine learning-based intrusion detection only focus on model construction and optimization. For example, Sahu et al. compared the performance of various linear and non-linear classifiers for NIDS using the KDD dataset [6], while Al-Maksousy compared deep neural networks (DNN) and various traditional machine learning (ML) models for NIDS on the same dataset, finding that DNN outperformed ML models in terms of accuracy, running time, and false positive rates [7]. Hu et al. proposed an incremental HMM training framework that incorporates a simple data pre-processing method for identifying and removing similar sub-sequences of system calls for HIDS [8]. This training strategy has been widely applied since it can save on the training cost (especially on large data) without noticeable degradation of intrusion detection performance [8]. Convolutional neural network (CNN) and recurrent neural network (RNN) have also been used in HIDS [9, 2]. Recently, Liu and Lang conducted a comprehensive survey on machine learning and deep learning methods for IDS [10]. Nevertheless, very few research studies have paid attention to data requirements, data

ISSN: 2366-1313

quality issues, and data quality assurance for IDS.

2 RELATED WORK

Intrusion detection aims to detect malicious activities or intrusions (break-ins, penetrations, and other forms of computer abuse) in a computer-related system (operating system or network system) [11]. An intrusion acts differently than the normal behavior of the system and, hence, the techniques used for anomaly detection can also be used for intrusion detection. Machine learning techniques used for intrusion detection can be divided into supervised learning and unsupervised learning. Whether the labeling of data is sufficient or not becomes the key criteria for selecting a machine learning technique. However, the detection performance of unsupervised learning methods is usually inferior to those of supervised learning methods [10]. Meanwhile, due to issues with how the data for intrusion detection typically flows (in a streaming fashion) and the data imbalance caused by low false alarm rates, the usage of machine learning techniques in intrusion detection is more challenging than other anomaly detection applications

Datasets for intrusion detection:

Many datasets have been created for intrusion detection [1, 2]. Datasets for NIDS mainly include information from the packet itself and aim at detecting the malicious activity in network traffic using the content of individual packets, while datasets for HIDS usually include information about events or system calls/logs on a particular system with the purpose of detecting vulnerability exploits against a target application or computer system.

Generally, the following properties are required when creating an IDS dataset: (1) Normal user behavior. The quality of an IDS is primarily determined by its attack detection rate and false alarm rate. Therefore, the presence of normal user behavior is indispensable for evaluating an IDS [31]. (2) Attack traffic. The attack types in different scenarios varies, so it is necessary to clarify the attacks in the IDS dataset. (3) Format. An IDS dataset can be in different formats such as packet-based, flow-based, host-based log files, etc. (4) Anonymity. Some of the information is anonymized due to privacy concerns, and this property indicates which attributes will be affected. (5) Duration. The recording time (e.g., daytime vs. night or weekday vs. weekend) of the dataset is indicated since a behavior might be regarded as an attack only when it occurs in a specific

ISSN: 2366-1313

duration. (6) Labeled. Labeled datasets are necessary for training supervised learning and semi-supervised learning models and for evaluating supervised learning, semi-supervised learning, and unsupervised learning models. (7) Other information, such as attack scenarios, network structure, IP addresses, recording environment, download URL, are also useful. Quality issues can easily appear in the above information. Those data quality issues, if not checked and eliminated appropriately, will greatly affect the intrusion detection performance. However, few studies have discussed the qualities of IDS datasets [3, 4] for machine learning, although data quality issues, such as duplication and imbalance, have been reported in the KDD dataset [6].

3 METHODOLOGY

Data preparation for machine learning-based IDS mainly includes four steps: (1) Selecting a data source or multiple data sources. (2) Collecting the data from the selected data source. (3) Labeling the data for training and testing. (4) Preprocessing the data as the model input. The workflow is shown in Figure 1. The data sources for HIDS and NIDS are different: data for HIDS can be collected from audit records, log files, the application program interface (API), rule patterns, and system calls, while data for NIDS is usually collected from the simple network management protocol (SNMP), network packets (TCP/UDP/ICMP), management information base (MIB), and router NetFlow records. Data can be collected from one data source or by integrating multiple data sources. The data source is the foundation of accessing high quality data. Once the data source is confirmed, additional information should be collected, such as metadata, format, duration, etc., for further analysis. The data collecting procedure should be well designed to ensure the data quality. For example, when collecting sequential events, they should be correctly organized by their order. Data labeling is an essential step for supervised machine learning-based IDS. Most existing machine learning algorithms make the assumption that the training data feeding the algorithms is accurate (has no errors). However, errors in label data entry, lack of precision in expert judgment, and imbalanced data distribution in different categories during the process of labeling the training examples can impact the predictive accuracy of the classification algorithms [48]. Data preprocessing aims at removing the outliers, cleaning the data, extracting the useful features, and splitting the data for training and

ISSN: 2366-1313

test. This process, if handled inappropriately, will cause data quality issues, such as data sparsity and bias, and overlapping between training and test, which can also reduce the performance of the machine learning algorithms.

"Timeliness" (also called "currency") refers to the extent to which the age of the data [56] is appropriate for the IDS task. Timeliness is an important factor to affect the performance of machine learning models since new types of attacks are emerging constantly, and some existing datasets, such as DARPA and KDD99, are too old to reflect these new attacks. Although the ADFA dataset contains many new attacks, it cannot be considered as comprehensive. For that reason, testing of machine learning models for IDS using DARPA, KDD99, and ADFA datasets does not offer a real evaluation and could result in inaccurate claims for their effectiveness [2]. Ideally, datasets should include most of the common attacks and correspond to current network environments [10].

"Variety" concerns the coverage of the instances on the selected features. For example, the KDD-Cup99 dataset has 41 features, and it is supposed to be a normal distribution in the selected features with known mean and standard deviation in the real world. Otherwise, it will induce the data sparsity issue. Moreover, to improve the robustness in machine learning models, the instances in the validate data and test data should have enough variety to test the training model. Variety is considered as a subset of comprehensiveness in the scenario of constructing a machine learning system for intrusion detection.

"Data consistency" refers to the validity and integrity of data representing real-world entities [5]. It aims to detect errors, such as inconsistencies and conflicts in the data, typically identified as violations of data dependencies [7]. For example, the system call for HIDS should be represented to ensure the sequential order, and the value of an attribute's domain (feature) should be constrained to the range of admissible values.

4 EXPERIMENT & RESULTS

Since the goal of this case study is to develop a host-based intrusion detection system (HIDS), we conduct machine learning experiments on the UNM, MIT, and ADFA-LD datasets. A detailed introduction of these datasets can be found in our GitHub repository.

ISSN: 2366-1313

Regardless of the slight difference in ADFA and UNM data formats, we use similar data cleaning and augmentation techniques to create a dataset for each class. Processing data for pre-trained language models vectors, such as BERT [6] and GPT-2 [7], is similar to normal machine learning algorithms. We use tokenizer to parse data into system call sequences with a length of six. Since the UNM dataset contains system calls from concurrently running processes, we group them by PID to ensure their sequential order. On the other hand, as the ADFA-LD dataset is already organized by different processes, and there is no PID provided, we do not need to group system calls together. Once the data is in order, we tokenize them into a sequence of six. By tokenizing into a sequence of 6-grams, we increase the amount of data for training as well as testing purposes. In addition, the number of features will decrease when a trace is tokenized into smaller chunks, and this will increase the efficiency of training as well as testing performances

We clean the data by removing rows or sequences that appear in both normal and intrusion data. This step draws distinctive characteristics between the two classes and effectively boosts the machine learning performance. A row with normal sequence is labeled 0, whereas the one with intrusion sequence is labeled 1. We use normal data and intrusion data from each dataset to create a sample pool. If it is imbalanced, we use the bootstrapping method to create a balanced sample of normal sequences and intrusion sequences. Then, we split the sample into training and testing sets in a 70-30 ratio. By training with only signature sequences from both classes, we increase the model accuracy and recall (true positive rate, TPR) as well as decrease its false positive rate (FPR).

Table 1: Model performance regarding accuracy, recall, precision, macro-F1, FPR, and AUC score on different HIDS

ISSN: 2366-1313

Dataset	Model	Accuracy	Rec
Synthetic Sendmail	K-means	0.63	0.6
	Logistic Regression	0.63	0.5
	SVM	0.73	0.5
	Neural Network	0.66	0.6
	Decision Tree	**0.98**	**1.0**
	Random Forest	**0.99**	**1.0**
	KNN	**1.00**	**1.0**
	Naïve Bayes	0.63	0.5
	BERT	**1.00**	**1.0**
	GPT-2 Network	**1.00**	**1.0**
Synthetic Ftp	K-means	0.20	0.0
	Logistic Regression	0.74	0.7
	SVM	0.79	0.6
	Neural Network	0.80	**0.8**
	Decision Tree	**0.99**	**1.0**
	Random Forest	**0.99**	**1.0**
	KNN	**0.99**	**1.0**
	Naïve Bayes	0.77	0.8
	BERT	**1.00**	**1.0**
	GPT-2 Network	**0.99**	**0.9**
Synthetic Lpr	K-means	0.55	0.1
	Logistic Regression	**0.97**	**0.9**
	SVM	**0.99**	**0.9**
	Neural Network	**0.97**	**0.9**
	Decision Tree	**0.99**	**1.0**
	Random Forest	**1.00**	**0.9**
	KNN	**1.00**	**1.0**
	Naïve Bayes	**0.96**	**1.0**
	BERT	**1.00**	**1.0**
	GPT-2 Network	**1.00**	**1.0**
	K-means	0.86	0.7
	Logistic Regression	**0.98**	**1.0**
	SVM	**1.00**	**1.0**
	Neural Network	**0.98**	**1.0**

5 CONCLUSION

Both the quality of a dataset and the capability of a model could contribute to the performance of a machine learning system. However, researchers usually under-value data work vis-a-vis model development [3]. In this article, we first discussed the data preparation workflow and data quality attributes for intrusion detection. Taking a HIDS as a case study, we then conducted experiments on 11 datasets using seven machine learning models and three deep learning models. Based on the experimental results, we propose the following conclusions: (1) Deep learning models, such as BERT and GPT-2, outperform the traditional machine learning models for intrusion detection since the former can encode the contextual information for sequential data (Figure 4). (2) Almost all of the algorithms achieved a better performance on the Synthetic Lpr, Live Lpr, Xlock, Live Named, Inetd, and Stide datasets than the others, indicating that the data quality of these datasets might be higher than the other datasets (Table 3). (3) Improving the data quality can enhance the performance of the machine learning performance in most of the situations (Figure 5). (4) The class imbalance issue in intrusion detection could be solved by using the bootstrapping technique to generate a balanced dataset in different categories. (5) Reputation, accuracy, and consistency are the data quality dimensions which yield high quality datasets for HIDS in this research. To assure data quality, we need to carefully check the data quality first. Tools such as the data validation system [9], BoostClean [5], and ActiveClean [6] can be used to detect and fix some potential data issues. However, these tools fail to connect data quality with machine learning performance. Therefore, it is necessary to conduct quantitative studies to verify the correlations between data

ISSN: 2366-1313

quality and machine learning performance. The second step is to improve the data quality. Different approaches can be used to improve data quality regarding different data quality dimensions. For example, to improve the correctness of a dataset, we may need to remove the label noises. To improve the variety of the dataset, we increase the unique data items in a dataset and make sure the distribution of the dataset follows the distribution of the population. To alleviate the data imbalance issue, we can use the bootstrapping technique to generate a more balanced dataset. Techniques such as transfer learning [5] and knowledge graph [7] have also been proved useful for data quality improvement.

References

[1] Abiodun Ayodeji, Yong-kuo Liu, Nan Chao, and Li-qun Yang. A new perspective towards the development of robust data-driven intrusion detection for industrial control systems. Nuclear Engineering and Technology, 2020.

[2] Ashima Chawla, Brian Lee, Sheila Fallon, and Paul Jacob. Host based intrusion detection system with combined cnn/rnn model. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 149–158. Springer, 2018.

[3] Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo."everyone wants to do the model work, not the data work": Data cascades in high-stakes ai. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, CHI '21, New York, NY, USA, 2021.

[4] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In Conference on fairness, accountability and transparency, pages 77–91, 2018.

[5] Haihua Chen, Jiangping Chen, and Junhua Ding. Data evaluation and enhancement for quality improvement of machine learning. IEEE Transactions on Reliability, pages 1–17, 2021.

[6] Abhijeet Sahu, Zeyu Mao, Katherine Davis, and Ana E Goulart. Data processing and model selection for machinlearning-based network intrusion detection. In 2020 IEEE International Workshop Technical Committee onCommunications Quality and Reliability (CQR), pages 1–6. IEEE, 2020.

[7] Hassan Hadi Al-Maksousy, Michele C Weigle, and Cong Wang. Nids: Neural network based intrusion

ISSN: 2366-1313

detection system. In 2018 IEEE International Symposium on Technologies for Homeland Security (HST), pages 1–6. IEEE, 2018.

[8] Jiankun Hu, Xinghuo Yu, Dong Qiu, and Hsiao-Hwa Chen. A simple and efficient hidden markov model scheme for host-based anomaly intrusion detection. IEEE network, 23(1):42–47, 2009.

[9] Nam Nhat Tran, Ruhul Sarker, and Jiankun Hu. An approach for host-based intrusion detection system design using convolutional neural network. In International Conference on Mobile Networks and Management, pages 116–126. Springer, 2017.

[10] Hongyu Liu and Bo Lang. Machine learning and deep learning methods for intrusion detection systems: A survey.applied sciences, 9(20):4396, 2019.

[11] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. ACM computing surveys (CSUR), 41(3):1–58, 2009.

[12] Shengchu Zhao, Wei Li, Tanveer Zia, and Albert Y Zomaya. A dimension reduction model and classifier foranomaly-based intrusion detection in internet of things. In 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech) , pages 836–843. IEEE, 2017.