Fall 12-1-2023

Investigating information systems vulnerabilities using machine learning algorithms

Shiyun Hao

Weihua Li

Xiong Zhang

Hao, S.Y., Li, W.H., Zhang, X. (2023). Investigating information systems vulnerabilities using machine learning algorithms. In Li, E.Y. *et al.* (Eds.) *Proceedings of The International Conference on Electronic Business, Volume 23* (pp. 401-408). ICEB'23, Chiayi, Taiwan, October 19-23, 2023

Investigating Information Systems Vulnerabilities Using Machine Learning Algorithms

Shiyun Hao [1]
Weihua Li [2]
Xiong Zhang [3,*]

*Corresponding author
[1] Graduate student, Beijing Jiao Tong University, Beijing, China, 21120609@bjtu.edu.cn
[2] Lecturer, Beijing Daxing District No. 1 Vocational School, Beijing, China, li_weihua0814@163.com
[3] Associate Professor, Beijing Jiao Tong University, Beijing, China, xiongzhang@bjtu.edu.cn

ABSTRACT

Companies and organizations equipped with IT infrastructure usually face security threats due to vulnerabilitiesin information systems. This paper aims to build models using intelligent algorithms to automatically identify vulnerability types and predict risk levels. We first collect reports from a Chinese vulnerability crowd-testing platform, then establish models by using textual representation technologies, shallow and deep learning algorithms. The experimental results show that the deep learning model with neural text representation could achieve better performance of vulnerability identification and risk level prediction. This research contributes to the information security literature and could help companies and organizations to more efficiently fix information systems vulnerabilities.

Keywords: Vulnerability, Information Security, Deep Learning, Text Representation, Identification and Prediction

INTRODUCTION

IT infrastructure and information systems are playing critical roles in operations and management among enterprises. Organizations in the age of cloud computing are facing the critical issue of integrating software (Zhang and Yue 2020). Cybersecurity issues such as data breaches due to information system vulnerabilities continue to be a major concern for firms. A report pointed out that vulnerability exploitation is one of the manners to do harm to data assets (Verizon 2022). Attackers exploit vulnerabilities to compromise information systems in enterprises. Thus, than ever before, firms are paying more attention to information systems vulnerabilities by heavily investing on human resource and R&D. Therefore, the key now is how to comprehensively understand the nature among various types of information systems vulnerabilities and their risk levels in order to fix vulnerabilities more efficiently. This study makes efforts towards this target by investigating vulnerability reports using machine learning algorithms.

This study constructed a dataset including reports of information security events from a well-known vulnerability crowd-testing platform in China. Such platforms could provide data for us to investigate information systems vulnerabilities. The platform improves the effectiveness of vulnerability mining, and increases the depth and breadth of mining of enterprise assets (Li and Zhao 2022).

The models we adopted in this study contain both traditional machine learning algorithms and artificial neural network. Different text representation methods are used for feature extraction of vulnerability reports. Both discrete representation and distributed representation are adopted and their performance are compared. We represent the text data as a matrix utilizing TF-IDF (Xue et al. 2019), BoW and N-gram. Besides, we denote the topics of each vulnerability report in a probability distribution by using the Latent Dirichlet Allocation model. Their topic distributions are extracted for subsequent experiments to classify the types of vulnerabilities and to predict risk levels of different kinds of vulnerabilities. In the experiment of artificial neural network, we use Word2Vec method to pre-process the text data of detailed descriptions of all vulnerabilities. Experiment results show that multilayer perceptron with neural text representation could achieve higher performance in both vulnerability identification and risk level prediction. This research has both theoretical and practical implications.

LITERATURE REVIEW

Existing studies have investigated vulnerabilities from perspectives of disclosure mechanisms, the interplay between firms and vulnerabilities, automatic detection of vulnerabilities, among others.

The mechanism of vulnerability disclosure is a complex issue. (Ahmed et al. 2021) proposed a comprehensive framework to examine the mechanisms of vulnerability disclosure from both market and non-market perspectives through a systematic literature review. This framework helps to comprehensively compare the two disclosure mechanisms.(Ransbotham et al. 2016) identified four types of relationships between digital vulnerabilities and ubiquitous IT: increased visibility, enhanced cloaking,

increased interconnectedness, and decreased costs. Ubiquitous computing makes various entities more vulnerable to attacks through these four mechanisms.

Some scholars explored the interplay between vulnerabilities and companies. (Zhuang et al. 2020) investigated the influence of awareness of a security vulnerability index on firm's security protection strategy, incentives and country-wide level of information technology (IT) development. Experimental results showed that countries in the Asia-Pacific region with high levels of ICT development are more responsive to cybersecurity vulnerabilities. (Wang et al. 2009) studied the impacts of cyber security on the firms with a focus on how disclosures of information security affected a firm's security management strategy. They found that the textual content of security risk could be used to predict future breaches. Some empirical studies have shown that information security incidents have negative impacts on company operations (Ye & Zhang 2021). Security vulneranilities, which are an important part of information security incidents, also have a negative impact on the market value of firms (Niu et al. 2022). Some studies have also explored the response of companys to network vulnerabilities, where the vulnerability risk levels and vulnerability types have a significant impact on the companies' ability to fix vulnerabilities (Hao et al. 2021). These studies provide management insights for company managers.

Recently, researchers examined the automation processes to detect vulnerabilities. (Grieco & Dinaburg 2018) proposed a proof-of-concept tool: a central development organizer, to optimize the detection tools of vulnerability, and help to specify initial values of parameters for a given vulnerability detection tool. (Spanos et al. 2017) adopted text mining techniques to analyze the vulnerability samples and confirmed the importance of vulnerability descriptions for vulnerability risk classification. They tested the effectiveness of TF-IDF representation against the simple word frequency representation. (Ruohonen 2017) identified attacks on open-source software vulnerabilities by using topic modeling tools and random forest classifiers. (Yitagesu et al. 2021) designed unsupervised word embedding models based on CBOW, combined it with negative sampling methods to extract features from security vulnerability. They proposed a method to assign part-of speech tags to tokens in detailed descriptions of vulnerabilities.(Wu et al. 2021) used the Word2Vec tool to convert tokens into vectors as inputs to the neural network for automatic vulnerability detection.(Zhang et al. 2020; Zhang et al. 2020) propose a general framework to understand vulnerabilities by using topic model and machine learning algorithms, and the framework helps to characterize the patterns and regularities of various type of vulnerabilities.

This study follows the research stream of automatic detection of vulnerabilities. But different from existing literature, this study focuses on the detailed description in textual data from vulnerability reports. We collected vulnerability reports from a well-known Chinese crowdsourcing testing platform. Various programming languages (e.g., C, C#), both Chinese and English comments, typos, URLs, screenshots, among others, exist in vulnerability reports, leading to difficulty in vulnerability mining. We combined text representation techniques with several state of art machine learning algorithms to automatically understand various vulnerabilities from a data-driven perspective. The work not only compares the ability of different text representation techniques on vulnerability report, but also compares the performance of machine learning model and artificial neural network in automatic identification of information system vulnerability.

DATA SET

The data source is an information system vulnerability crowd-testing platform in China. The crowd-testing platform is a meritocratic learning community (Zhang et al. 2015). The raw data of 39503 information security events from 2010 to 2016 was collected using a Python crawler. 20 different attributes were included in each event, such as: vulnerability ID, poster, vulnerability title, detailed description, associated company, submission time, fix time, disclosure time, vulnerability type and risk level, fix solution, and so on. *Vulnerability ID* is a public ID given by the platform for widely recognized information security vulnerabilities or vulnerabilities that have been exposed. *Poster* detected vulnerabilities and reported them into the platform. *Vulnerability Title* provides a brief summary of the vulnerability events. *Description* and *Solution* are the detailed description of vulnerability and the potential fixing solutions. *Vulnerability Type* and *Vulnerability Risk Level* indicate the type vulnerabilities belong to and how much damage vulnerability may cause. There are three levels of vulnerability risk in the dataset: high, middle, and low. Among these attributes, our focus is on the type, risk level, description, and solutions of vulnerability report. Table 1 shows the partial information of one vulnerability report.

Table 1: Partial Information of One Vulnerability Report

Attribute	Value
Vulnerability ID	2016-168160
Poster	Booooooom
Vulnerability Title	Internal API leakage from docker cloud service
Description	The place where the image is created supports the use of doc file for construction ...
Solution	Isolate the API of your own service from the user environment ...
Vulnerability Type	Design Defect/Logic Error
Vulnerability Risk Level	High

Associated Company	NetEase
Post Time	January 7, 2016
Fix Time	January 31, 2016
Release Time	February 22, 2016
Vulnerability Status	Vulnerability Post: 2016-1-7, waiting for confirmation of firm Vulnerability Fix: 2016-1-31, Firm confirmed, details were sent to firm, and fixed by NetEase. Release: 2016-2-22, after fixing, detail information released to public.
Firm Response	Thanks for your clear description.

The dataset contains lots of unstructured noisy data. The detailed description of the vulnerability includes source code, images, comments, etc. Source codes may be in different programming languages, e.g., SQL, Java, C, C#. The text content can be in English and Chinese. Besides, the report may also include typos, special symbols, URLs, figures, and so on. The figures are mainly screenshots of source code, result output, and user interfaces, which are not included in this study.

METHODOLOGY

This study analyzes vulnerability reports by combining textual representation methods, learning algorithms, and topic analysis. We extract features using different text representation methods and conduct experiments using algorithms of traditional machine learning and artificial neural network.

This study adopted two types of text representation. The first is statistical text representation, including word-level and document-level text representation techniques (Wawrzyński & Szymański 2021). Word-level text representation techniques could extract features from words, and these features can serve as the input to models of classification or prediction. This type of text representation techniques includes (1) TD-IDF assessing the importance of words in a document to a document or the importance of a document in a corpus, (2) Bag-of-Words models converting texts into vectors that represent the frequency of occurrence of a particular term, and (3) N-Gram models calculating the probability of occurrence of terms. The document-level representation technique uses Latent Dirichlet Allocation (LDA) to obtain document features and infer document topic distribution. We conduct experiments using features of both word and document levels.

The second type of text representation is neural text representation which is a word embedding model (Wawrzyński and Szymański 2021). Word2Vec is one of the language models that learn semantic knowledge in an unsupervised manner from a large number of texts. According to the difference between input and output, the word embedding method could be categorized into continuous bag-of-words model (CBOW, predicting the current value by context) and Skip-Gram (SG, use the current word to predict the context). To speed up the training process, we adopt training mechanisms of Hierarchical SoftMax (HS) and Negative Sampling (NS) combined with CBOW and SG separately. For the abovementioned two models and two training mechanisms, we would try out all the four combinations in this study.

Text classification can be accomplished by utilizing traditional machine learning algorithms or deep learning algorithms (Li et al. 2022). Traditional Machine learning algorithm adopted in our experiment includes Logistic Regression, Classification and Regression Trees, Random Forest, Gradient Boosting Decision Tree, and Support Vector Machines. The deep learning algorithm adopted in this study is Artificial Neural Network.

EXPERIMENTS AND RESULTS

This section first describes the data preparation process, then shows how we conduct feature extraction, experimental process, and finally present the results of vulnerability type identification and risk level prediction. Figure 1 presents the whole process of the experiment.

Data Preparation

We preprocess vulnerability reports in the following steps.

(1) Remove records missing detailed description. 39503 records in 16 types of vulnerabilities were collected in the crowdsourcing platform. Among them, 86 records missing detailed description were removed. We performed statistical analysis for the detailed vulnerability description during the initial processing. The average count of characters for the detailed description of vulnerabilities is about 859.52.

(2) Text data cleaning. The dataset is a mixture of both Chinese and English texts. We delete the URLs, numbers, punctuation, the numbered serial numbers with circles, and space among the text. We convert all uppercase letters to lowercase letters. Then, the texts of vulnerability detailed description were sliced by using the jieba module and stop words are removed. The stop-word list is constructed by combining the commonly used Chinese stop-word lists (from the Harbin Institute of Technology, Baidu, Si Chuan University, and CN) and English long stop-word lists.

(3) Rows with empty values and records with extremely short detailed descriptions are removed.

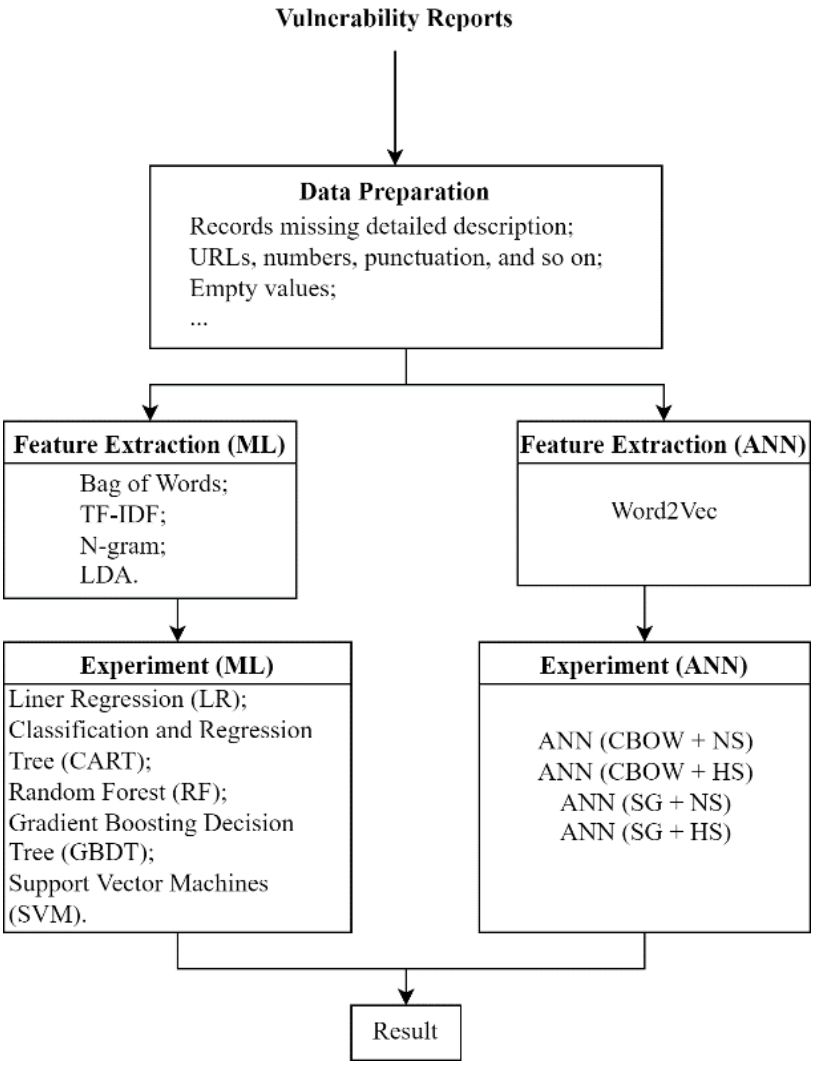

Figure 1: General Framework of Experiment

Finally, the dataset includes 30123 valid records. We present the count of vulnerabilities in terms of type and risk level in Table 2.

Table 2: Summary of Various Types of Vulnerabilities

Vulnerability Type	*Risk Level*			*All*
	High	*Middle*	*Low*	
Cross-site Scripting (XSS)	965	1078	742	2785
SQL Injection	5533	1573	311	7417
Weak Password	1696	575	148	2419
Successful Intrusion Event	821	108	66	995
Sensitive Information Disclosure	1797	775	420	2992
File Operation Vulnerability	1476	406	108	1990
Configuration Error	1175	373	196	1744
Design Defect/Logic Error	2797	1062	590	4449
Remote Code Execution	2068	435	91	2594
Unauthorized Access/Permission Bypass	1586	859	293	2738
All	19914	7244	2965	30123

Among all types of vulnerabilities, SQL injection, Design Defect/Logic Error and Sensitive Information Disclosure occur more frequently than other types of vulnerabilities. They account for almost 50% of the dataset.

Feature Extraction

For traditional machine learning methods, features are extracted in three dimensions: BOW with n-grams, TF-IDF with n-grams, and document topics. The Bag-Of-Words model does not consider word order in sentences, it transforms a sentence into a vector representation based on the occurrences of the words in that sentence. In order to highlight the role of keywords in detailed descriptions, we also adopt TF-IDF method. Besides, we used a hybrid pattern of unigram and bigram combined with the text representation methods at the word level, to incorporate word order effects. Bi-gram is a maximum probability partition, which considers not only itself but also its antecedents when calculating the word probability. We set the number of topics in vulnerability reports to be 40, which is fairly large number to cover most topics in reports.

For ANN, we adopt the Word2Vec embedding method for text representation and feature extraction.

Experiment

We adopt two text representation techniques and various classification algorithms to automatically identify the types of vulnerabilities and evaluate risk levels. First, we establish the models by combining statistical text representation techniques and traditional machine learning algorithms: LR, CART, RF, GBDT, and SVM. All the experiments were conducted using 5-fold cross validation. The whole dataset is randomly partitioned into 5 subsets, out of which 4 subsets are used to train models to identify vulnerability types and to evaluate risk levels, the last subset is used to evaluate the performance of the trained models. The performance is measured using the metric of area under the curve (AUC).

Then, we established the models by combining neural text representation techniques and ANN. When training ANN models, we set the dimension of the eigenvector to 100 and ignored words with fewer than 5 occurrences. We added a layer of Dropout to avoid overfitting. The activation function between networks is ReLU. The activation function for the last layer of the network is SoftMax, since the task is multi-categorized. We evaluate the performance of all the models in AUC.

Vulnerability Type Identification

The performance of proposed identifiers is presented in Table 3.

Table 3: Performance of Vulnerability Type Identification

Vulnerability Type	Statistical Text & Machine Learning					Neural Text & ANN			
	LR	CART	RF	GBDT	SVM	ANN (CBOW+NS)	ANN (CBOW+HS)	ANN (SG+NS)	ANN (SG+HS)
Cross-site Scripting (XSS)	0.9656	0.8337	0.9709	0.974	0.9602	0.9835	0.9813	0.9823	0.9804
SQL Injection	0.9623	0.847	0.9632	0.9632	0.9575	0.9706	0.9722	0.9729	0.9714
Weak Password	0.9304	0.7083	0.9333	0.9296	0.923	0.9412	0.9431	0.9472	0.9431
Successful Intrusion Event	0.7929	0.5275	0.8069	0.8092	0.7858	0.8454	0.8507	0.8599	0.8569
Sensitive Information Disclosure	0.7950	0.6002	0.8236	0.8284	0.8052	0.8511	0.8514	0.8592	0.8559
File Operation Vulnerability	0.8996	0.6665	0.919	0.9171	0.8933	0.9254	0.9283	0.9299	0.938
Configuration Error	0.7981	0.5746	0.7932	0.8054	0.788	0.8221	0.8309	0.8388	0.8319
Design Defect/ Logic Error	0.8706	0.677	0.8863	0.8795	0.8700	0.9024	0.9082	0.9074	0.9068
Remote Code Execution	0.9082	0.7433	0.9299	0.9241	0.8996	0.9464	0.9459	0.9457	0.9445
Unauthorized Access/ Permission Bypass	0.8174	0.5945	0.8367	0.844	0.8224	0.8548	0.8569	0.8630	0.8574
Average	0.8740	0.6773	0.8863	0.8875	0.8705	0.9043	0.9069	0.9106	0.9086

From Table 3, the overall classification results of vulnerability types are acceptable. We find that the identification rate is high in Cross-site Scripting, SQL Injection, Weak Password, File Operation Vulnerability and Remote Code Execution. The identification performance of these types of vulnerabilities is higher or close to 90%. The average AUC classification performance for Design Defect/Logic Error is 90.68%, which is close to the average AUC for all vulnerability categories. On the other hand, the identification rate is low in Successful Intrusion Event and Configuration Error. The small sample size of these two types of vulnerabilities may be the reason for low identification rate. Follow-up studies can construct a more balanced dataset to train the identifier. In addition, the average identification performance of Sensitive Information Disclosure and Unauthorized Access/Permission Bypass is around 80%.

The relatively low classification performance of these four types of vulnerabilities may be due to the unbalance dataset and the complexity of vulnerabilities. Successful Intrusion Event includes a variety of intrusion methods, such as: intrusion of malicious files, successful intrusion due to wrong use of editing tools, etc. Various intrusion methods lead to the similarity of this

vulnerability category with other vulnerability types, thus it is difficult to identify. Configuration Error are mainly the result of improperly configured system, service operations and maintenance. Thus, it is usually exploited by attackers, since it usually involves numerous human factors or defects in the information system itself. Therefore, the variety of features results in low classification performance. Unauthorized Access/Permission Bypass means that attackers without authentication can still remote login servers. These vulnerability records usually contain illegal manipulation of databases or website directories, as well as leakage of sensitive information. Therefore, these records can also be classified as Sensitive Information Disclosure.

In the first identification model, the identification rate is relatively higher in RF and GBDT, but is lowest in CART. The identification rate of CART is 67.7%, but the identification rates of RF and GBDT are above 88%. In terms of model characteistics, it is more difficult for CART to predict continuous text. RF and GBDT improve CART by assembling weak classifiers into more powerful ones. Thus, the results of RF and GBDT are significantly improved. LR and SVM are common linear models. Their identification rates are more than 87%, slightly lower than GBDT and RF. In the second identification model, all the identification rates of each identifier are higher than 0.9. SG+NS could achieve the highest identification rate. SG could perform better than CBOW model for the reason that SG is more suitable to analyze long texts. Overall, ANN could outperform traditional machine learning models when identifying information systems vulnerabilities.

Risk Level Prediction

Now we move our focus to evaluate the risk levels for all types of information systems vulnerabilities. The performance of risk level prediction is presented in Table 4.

Table 4: Performance of Risk Level Prediction

Risk Level	Statistical Text & Machine Learning					Neural Text & ANN			
	LR	CART	RF	GBDT	SVM	ANN (CBOW+NS)	ANN (CBOW+HS)	ANN (SG+NS)	ANN (SG+HS)
High	0.7418	0.5794	0.7250	0.7363	0.7198	0.7417	0.7431	0.7445	0.7441
Middle	0.5914	0.5402	0.6444	0.6638	0.6340	0.6728	0.6719	0.6705	0.6715
Low	0.7532	0.5686	0.7243	0.7553	0.7351	0.7628	0.7627	0.7656	0.7727
Average	0.6955	0.5627	0.6979	0.7185	0.6963	0.7257	0.7259	0.7269	0.7294

Table 4 shows that the prediction rates of low risk level and high risk level vulnerabilities are higher than that of middle risk level vulnerabilities. The possible reason of a low prediction rate of middle risk level vulnerabilities is that its detailed description is relatively ambiguous, especially compared to those of high and low risk level vulnerabilities. Take Cross-site Scripting (XSS) as an example. There exist significant difference in the description between a high risk XSS and a low risk XSS, which are caused by poor data filtering. High risk XSS can lead to Web page hanging horse, identity theft, XSS worm attacks and so on. Some attackers utilize XSS to steal cookies, and view users' privacy. Some high risk XSSs attacks can even hijack a user's web behavior to monitor their browsing history, data they send and receive, etc. For low risk XSS, posters mostly describe the location of the vulnerability to warn companies, such as headers of logs or comments. The description of low risk vulnerabilities is significantly simpler.

On the other hand, the description of middle risk vulnerabilities is relatively ambiguous. Let's take, Baihe, an online dating platform with a large number of daily visitors, as an example. The poster described the impact of a middle risk level XSS in Baihe as: *"if the malicious code inserted, its spreading impact is still not insignificant"*. Vague description like those leads to the difficulty to define vulnerabilities in middle risk level. This may lead to a relatively lower evaluation performance.

From algorithm point view, GBDT performs better in risk level prediction. Its performance was similar to that of ANN. LR performs better when predicting low and high-risk levels. The performance of RF, GBDT and SVM in risk level prediction is more stable. In contrast, CART performs worst in risk level prediction tasks. CART has high computational complexity and is not suitable for high-dimensional sparse features. Since there exist a large number of features in our experiments, it takes a lot of time to train each regression tree. This may explain the low performance of CART.

For ANN, SG could achieve higher prediction rate than CBOW since SG is suitable to analyze long documents. Besides, HS slightly outperforms NS since HS is good at pre-training of infrequent words. Overall, ANN outperforms traditional machine learning models by about 2% in AUC score.

CONCLUSION AND FUTURE DIRECTION

This research adopts intelligent models and various text representation techniques to comprehensively understand information systems vulnerabilities. First, we collected reports of information system vulnerabilities from a crowd-testing platform in China. Then we conducted experiments to automatically identify vulnerability types and predict their risk levels. Our experimental results show that the combination of ANN and neural text representation could outperform other state of art algorithms. Therefore, this research is of significant importance to both vulnerability literature and cyber security practice.

Vulnerability reports may include image and screenshots demonstrating codes, technical problems, among others, associated with information systems vulnerabilities. Future research could adopt image-understanding techniques to improve the performance of models to understanding vulnerabilities.

ACKNOWLEDGEMENT

This research is supported by grants from the National Natural Science Foundation of China (Grant 71801014) and Beijing Social Science Foundation (Grant 17GLC069). Xiong Zhang (xiongzhang@bjtu.edu.cn) is the corresponding author.

REFERENCES

Ahmed, A., Deokar, A., and Lee, H. C. B.（2021). "Vulnerability Disclosure Mechanisms: A Synthesis and Framework for Market-Based and Non-Market-Based Disclosures," *Decision Support Systems* (148). https://doi.org/10.1016/j.dss.2021.113586

Grieco, G., and Dinaburg, A. (2018). "Toward Smarter Vulnerability Discovery Using Machine Learning," *Proceedings of the ACM Conference on Computer and Communications Security*, Association for Computing Machinery, October 15, pp. 48–56. https://doi.org/10.1145/3270101.3270107

Yang, H., Zhang, J. F., and Zhang, X. (2021). "Network Vulnerability and Enterprises' Response: The Preliminary Analysis," In Chan, Y. et al. (Eds.) *Proceedings of Twenty-Seventh Americas Conference on Information Systems (AMCIS 2021)*, Montreal, Canada, August 9-13. https://aisel.aisnet.org/amcis2021/info_security/info_security/22

Li, Q., Hao, P., Li, J. X., Xia, C. , Yang, R., Sun, L., Yu, P. S., and He, L. (2022). "A Survey on Text Classification: From Traditional to Deep Learning," *ACM Transactions on Intelligent Systems and Technology* (13:2), Association for Computing Machinery, pp. 1–41. https://doi.org/10.1145/3495162

Li, Y., and Zhao, L. (2022). "Collaborating with Bounty Hunters: How to Encourage White Hat Hackers' Participation in Vulnerability Crowdsourcing Programs through Formal and Relational Governance," *Information and Management* (59:4). https://doi.org/10.1016/j.im.2022.103648

Niu, X., Li, W., Zhang, J., Zhang, X., Chen, J., and Xu, H. (2022). "The Impact of Information Systems Vulnerability Announcements on Firms' Market Value," In Li, E.Y. et al. (Eds.) *Proceedings of the 22nd International Conference on Electronic Business (ICEB 2022)*, pp. 691–700. https://aisel.aisnet.org/iceb2022/70/

Ransbotham, S., Fichman, R. G., Gopal, R., and Gupta, A. (2016). "Ubiquitous IT and Digital Vulnerabilities," *Information Systems Research* (27:4), pp. 834–847. https://doi.org/10.1287/isre.2016.0683

Ruohonen, J. (2017). "Classifying Web Exploits with Topic Modeling," *Proceedings of 2017 28th International Workshop on Database and Expert Systems Applications (DEXA)* (Vol. 2017-August), IEEE, Lyon, France, September 28, pp. 93–97. http://ieeexplore.ieee.org/document/8049693/

Spanos, G., Angelis, L., and Toloudis, D. (2017). "Assessment of Vulnerability Severity Using Text Mining," in *21st Pan-Hellenic Conference on Informatics* (Vol. Part F132523), Association for Computing Machinery, September 28..

Verizon. 2022. "DBIR Data Breach Investigations Report." https://www.verizon.com/business/en-gb/resources/reports/dbir/ (accessed 1 February 2023).

Wang, T., Ulmer R. J., and Kannan, K. N. (2009). "The Association between the Disclosure and the Realization of Information Security Risk Factors," *Information Systems Research* (24:2). https://doi.org/10.1287/isre.1120.0437

Wawrzyński, A., and Szymański, J. (2021). "Study of Statistical Text Representation Methods for Performance Improvement of a Hierarchical Attention Network," *Applied Sciences (Switzerland)* (11:13), MDPI AG. https://doi.org/10.3390/app11136113

Wu, T., Chen, L., Du, G., Zhu, C., and Shi, G. (2021). "Self-Attention Based Automated Vulnerability Detection with Effective Data Representation," *Proceedings of 2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking*, New York City, NY, USA: IEEE, December 22, pp. 892–899. https://ieeexplore.ieee.org/document/9644777

Zhang, X., Xie, H., Yang, H., Shao, H., and Zhu, M. (2020). "A General Framework to Understand Vulnerabilities in Information Systems," *IEEE Access* (8), Institute of Electrical and Electronics Engineers Inc., pp. 121858–121873. https://ieeexplore.ieee.org/document/9130665

Zhang, X., Shao, H., Zhu, M., and Zhang, R. (2020). "Towards Understanding Vulnerability in Information Systems: The Topic Modeling Perspective," *Proceedings of the Twenty-Fourth Pacific Asia Conference on Information Systems (PACIS 2020)*, Dubai, UAE, June 20-24. https://aisel.aisnet.org/pacis2020/242/

Xue, J., Liu, K., Lu, Z., and Lu, H. (2019). "Analysis of Chinese Comments on Douban Based on Naive Bayes," *Proceedings of the 2nd International Conference on Big Data Technologies (ICBDT 2019)*, Association for Computing Machinery,August, pp. 121–124. https://doi.org/10.1145/3358528.3358570

Ye, R., and Zhang, X. (2021). "Information Security and Firms' Market Value: The Preliminary Analysis," *Proceedings of the 21st International Conference on Electronic Business (ICEB 2021)*, Nanjing, China, pp. 235–243. https://aisel.aisnet.org/iceb2021/11/

Yitagesu, S., Zhang, X., Feng, Z., Li, X., and Xing, Z. (2021). "Automatic Part-of-Speech Tagging for Security Vulnerability Descriptions," *Proceedings of ACM 18th International Conference on Mining Software Repositories, MSR 2021*, Institute of Electrical and Electronics Engineers Inc., May 1, pp. 29–40. https://ieeexplore.ieee.org/document/9463114

Zhang, X., Tsang, A., Yue, W. T., and Chau, M. (2015). "The Classification of Hackers by Knowledge Exchange Behaviors," *Information Systems Frontiers* (17:6), Springer New York LLC, pp. 1239–1251. https://doi.org/10.1007/s10796-015-9567-0

Zhang, X., and Yue, W. T. (2020). "Integration of On-Premises and Cloud-Based Software: The Product Bundling Perspective," *Journal of the Association for Information Systems* (21:6), Association for Information Systems, pp. 1507–1551. https://aisel.aisnet.org/jais/vol21/iss6/6/

Zhuang, Y., Choi, Y., He, S., Leung, A. C. M., Lee, G. M., and Whinston, A. (2020). "Understanding Security Vulnerability Awareness, Firm Incentives, and ICT Development in Pan-Asia," *Journal of Management Information Systems* (37:3), Routledge, pp. 668–693. https://doi.org/10.1080/07421222.2020.1790185