RESEARCH ARTICLE OPEN ACCESS

SOFTWARE VULNERABILITY CLASSIFICATION USING NEURAL NETWORKS

Ms. N.Zahira Jahan, MCA., M.Phil *, Mr. K.Gokulkrishnan**
*(Associate Professor, Department of Computer Applications,
Nandha Engineering College (Autonomous),
Erode, Tamil Nadu, India
Email: zahirajahan1977@gmail.com)
** (Final MCA, Department of Computer Applications,
Nandha Engineering College (Autonomous),
Erode, Tamil Nadu, India
Email: k.gokulkrishnan97@gmail.com)

----------**********----------

Abstract:

Software vulnerabilities are raising the security risks. If any vulnerability is oppressed due to a malicious attack, it will compromise the system's safety. In addition, it may create catastrophic losses. So, automatic classification methods are required to manage vulnerability in software, and then security performance of the system will be improved. It will also mitigate the risk of system being attacked and damaged. In this project, a new model has been proposed with name automatic vulnerability classification model (IGTF-DNN) Information Gain based on Term Frequency - Deep Neural Network. The model is generated using information gain (IG) which is based on frequency-inverse document frequency (TF-IDF), and deep neural network (DNN): TF-IDF is used to calculate frequency/weight of words taken from vulnerability description; IG is used to select features to gather optimal set of feature words. Then neural network model is used to construct an automatic vulnerability classifier to achieve effective vulnerability classification. The National Vulnerability Database of the United States has been taken to test this new model's effectiveness. By comparing with KNN, this TFI-DNN model has achieved better performance in evaluation. Software vulnerabilities are raising the security risks. If any vulnerability is oppressed due to a malicious attack, it will compromise the system's safety. In addition, it may create catastrophic losses. So, automatic classification methods are required to manage vulnerability in software, and then security performance of the system will be improved. It will also mitigate the risk of system being attacked and damaged. In this project, a new model has been proposed with name automatic vulnerability classification model (IGTF-DNN) Information Gain based on Term Frequency - Deep Neural Network. The model is generated using information gain (IG) which is based on frequency-inverse document frequency (TF-IDF), and deep neural network (DNN): TF-IDF is used to calculate frequency/weight of words taken from vulnerability description; IG is used to select features to gather optimal set of feature words. Then neural network model is used to construct an automatic vulnerability classifier to achieve effective vulnerability classification. The National Vulnerability Database of the United States has been taken to test this new model's effectiveness. By comparing with KNN, this TFI-DNN model has achieved better performance in evaluation

. *Keywords* — Neural Network, security, classification model.

----------**********----------

I. INTRODUCTION

Due to information technology's fast development, its impacts to industries by application of the Internet and computers are powerful. Not only, they brought convenience, but also huge risks and hidden dangers at the same time. With the improvement of digitalization level of industries, information security problems have become increasingly outstanding. Vulnerabilities are nothing but software/hardware defects of system being illegitimately exploitable made by unauthorized people. As soon as vulnerability of information system is exploited by suspicious attack, the information system's security will be at great risk. It may even create inestimable consequences. In 2017, Windows system vulnerabilities are exploited by hackers to expose 100,000 organizations around the world to Bit coin ransom ware. Again in the same year, Microsoft released a total of 372 vulnerability patches for Office. Hackers make use of office vulnerabilities to conduct Advanced Persistent Threat (APT) attacks, spread ransom ware, botnets and so on. Nowadays, the count and variety of vulnerabilities are gradually increasing, so that the analysis and management of software vulnerabilities are becoming more important. If the vulnerability can be classified and managed with effectiveness, it may not only enhance the efficiency of vulnerability recovery and management, but also diminish the risk of systems being attacked and collapsed, which is crucially important for security performance of systems. As software security vulnerabilities play a major role in cyber-security assault, more and more researches on vulnerability classification are conducted by applicable security researchers.

II. RELATED WORK

Even though these machine learning algorithms have achieved hopeful results in many fields, because of the huge amount of vulnerability data with short description, generated word vector space handed the characteristics of high dimension and sparse. These machine learning algorithms are not very much effective dealt with high as well as sparse problems. Meanwhile, they pay no attention to particular vulnerability information and so the classification accuracy is not elevated. In recent years, deep learning found application in variety of fields and has achieved triumph, such as the speech and image recognition field and there, the error rate in speech recognition is lowered by 20 – 30 percent [1]. The error rate in ImageNet evaluation task is lowered by 26 – 15 percent [2]. Deep learning also has a important impact in the natural language filed [3], [4].

Jo et al. [5] studied the classification problems in the natural language field, and they applied convolutional neural networks (CNN) and recurrent neural networks (RNN) for large-scale text classification and achieved success.

Aziguli et al. [6] introduced a new text classifier using DNN model to progress the computational performance of processing large text data along with mixed outliers. Therefore, to better deal with the high and sparse word vector space and thereby take benefits of automatic feature extraction by deep learning, this paper introduces an automatic vulnerability classification model IGTF-DNN based on term frequency-reverse document frequency (TF-IDF), information gain (IG) and deep neural network (DNN).

In the model, IGTF algorithm is first used to grab the feature of description text and reduce the generated high-dimensional word vector space dimension. Then a DNN neural network model (based on deep learning) is constructed. The model was trained and tested with vulnerability data taken from National Vulnerability Database (NVD). The test results showed that the automatic vulnerability classification model in this paper effectively improves the performance of vulnerability classification.

III. EXSISTING SYSTEM

A. TF-IDF (Term Frequency/Inverse Document Frequency) is a common weighted technology which is found out based on statistical methods [17]. For example, consider there are a set of documents and each document contains a number of

terms/words. It is defined that the word I's importance in document j as follows.

$$Tfij = ni,j / \sum k\ nk,j \qquad (1)$$

Where both I and j are positive integers, ni,j denotes the term I's frequency in document j.

The IDF formula is as follows.

$$idfi = \log (|F| / |\{ j : ti \square dj \}|) \qquad (2)$$

Where |F| is the total number of documents in corpus, fj is the jth document, and |{j:ti ∈fj| is the number of documents containing the term ti.
The TF-IDF formula is as follows.

$$TF{-}IDF = tfij * idfi \qquad (3)$$

TF-IDF is used to measure the terms' importance word to a document in the document set o rin a corpus. The terms' importance increases proportionally with number of times it appears in the document, but also decreases inversely with frequency it appears in corpus.

B. INFORMATION GAIN (IG) refers to that, if a feature X in class Y is known already, information uncertainty of class Y decrease, and so reduced uncertainty degree will reflect importance of feature X to class Y. Set the training data set to D, |D| shows the count of samples in D. Suppose there are K classes Ck, k = 1, 2, ⋯ , K |CK| is the count of samples fit in to class Ck. ∑Ck=1 K |CK| = |D|. If feature A has n different values {a1, a2, ⋯, an}, D is segmented into 'n' sub groups according to feature A values, represented as D = (D1, D2, ⋯ , Dn), where |Di| is the samples count in Di, ∑i=1 n |Di| = |D|. The samples set fit into class Ck in Di is Dik, Dik = Di ∩ Dk, |Dik| is the samples count of Dik.

The empirical entropy H (D) of data set D is calculated as follows. (4)

The empirical conditional entropy H(D|A) of feature A for dataset D is calculated as follows (5)

The information gain calculation formula for each feature is as follows. (6)

Based on feature selection method of information gain criterion, each feature's information gain is measured, and the features with larger IG value are selected.

IV. PROPOSED SYSTEM

In proposed system, all the existing methodology is carried out. Deep neural network is used to automatically classify the vulnerability document content in the vulnerability type with back propagation technique so that hidden layer weights are readjusted with effective values.

The following modules are present in the project.

1. TERMFREQUENCY–INVERSEDOCUMENT FREQUENCY
2. INFORMATION GAIN
3. FEATURE WORDS EXTRATION
4. OPTIMIZATIONS USING DNN

1. TF-IDF (Term Frequency/Inverse Document Frequency)

It is a common weighted technology which is found out based on statistical methods [17]. For example, consider there are a set of documents and each document contains a number of terms/words. It is defined that the word I's importance in document j as follows.

$$Tfij = ni,j / \sum k\ nk,j \qquad (1)$$

Where both I and j are positive integers, ni,j denotes the term I's frequency in document j.

The IDF formula is as follows.

$$idfi = \log (|F| / |\{ j : ti \square dj \}|) \qquad (2)$$

Where |F| is the total number of documents in corpus, fj is the jth document, and |{j:ti ∈fj| is the number of documents containing the term ti.

The TF-IDF formula is as follows.

$$TF{-}IDF = tfij * idfi \quad (3)$$

TF-IDF is used to measure the terms' importance word to a document in the document set o rin a corpus. The terms' importance increases proportionally with number of times it appears in the document, but also decreases inversely with frequency it appears in corpus.

2. *Information Gain (IG)*

It refers to that, if a feature X in class Y is known already, information uncertainty of class Y decrease, and so reduced uncertainty degree will reflect importance of feature X to class Y. Set the training data set to D, |D| shows the count of samples in D. Suppose there are K classes Ck, k = 1, 2, ⋯ , K |CK| is the count of samples fit in to class Ck. ∑ Ck=1 K |CK| = |D|. If feature A has n different values {a1, a2, ⋯, an}, D is segmented into 'n' sub groups according to feature A values, represented as D = (D1, D2, ⋯ , Dn), where |Di| is the samples count in Di, ∑i=1 n |Di| = |D|. The samples set fit into class Ck in Di is Dik, Dik = Di ∩ Dk, |Dik| is the samples count of Dik.

The empirical entropy H (D) of data set D is calculated as follows (4)

The empirical conditional entropy H(D|A) of feature A for dataset D is calculated as follows (5)

The information gain calculation formula for each feature is as follows (6)

Based on feature selection method of information gain criterion, each feature's information gain is measured, and the features with larger IG value are selected.

3. *Feature Words Extraction*

In this module, the following algorithm is worked out.

Input:

Word list(word_list) formed by term document matrix and stop word filtering.

Output:

Feature word set (feature_words).

1) Traversing each word in the word_list.

2)Word frequency statistics for word_list, stored in the doc_frequency list.

3) Traversing the word frequency list doc_frequency.

4) Calculate the TF value of each word according to (1) and store it in the word_tf dictionary.

5) Calculate the IDF value of each word according to (2) and store it in the word_idf dictionary.

6) Calculate the TF-IDF value of each word according to (3) and store it in the word_tf_idf dictionary.

7) The word set is sorted in descending according to the TF-IDF value.

8) Select the first n words as an important feature set.

9) Save important words in the feature list (features

10) Traverse features_vocabSet, divide features_vocabSet and store the subset into the subDataSet.

11) Calculate probability of subDataSet.

12) Calculate the empirical conditional entropy of each word according to (4) and (5) and store it in newEntropy.

13) Calculate the IG value of each word according to (6).

14) Save each word and the corresponding IG value in the dictionary.

15) The word set is sorted descending by IG value.

16) Select the first m words as features and store them in the feature_words.

17) Return feature_words.

4. *Optimizations Using DNN*

In this module, DNN is used which consists of one input layer, multiple hidden layers and one output layer, whose input is the feature vector of instance and output is the category of instance. It mainly includes two propagation processes, forward propagation and back propagation. The propagation process is in Algorithm 2.

V. CONCLUSION

In order to better analyze and manage vulnerabilities according to their belonging classes, improve the security performance of the system, and reduce the risk of the system being attacked and

damaged, this paper applied deep neural network to software vulnerability classification. The analysis of the method and construction process of TFI and DNN are discussed in detail. The comparison is made with the vulnerability classification model TFI-DNN to TFI-SVM, TFI-Naïve Bayes and TFI-KNN on the NVD dataset. The results show that the proposed TFI-DNN model outperforms well in preparing weights and biases. And it is superior to general TF-IDF on comprehensive evaluation indexes. The work in this paper shows the effectiveness of TFI-DNN in vulnerability classification, and provides a basis for our future research using the benchmark vulnerability dataset.

REFERENCES

[1] Gruber, D., 2012. Product Quality and International Price Dynamics, s.l.: s.n. Heathcote, J. & Perri, F., 2002. Financial autarky and international business cycles. Journal of Monetary Economics, 49(3), pp. 601-627.

[2]Helbling, T., Huidrom, R., Kose, M. A. &Otrok, C., 2011. Do credit shocks matter? A global perspective. European Economic Review, Volum 55, pp. 340-353. Humphrey, J., 2009. Are exporters in Africa facing reduced availability of trade finance?. IDS Bulletin, 40(5), pp. 28-37.

[3]Iacovone, L. &Zavacka, V., 2009. Banking Crises and Exports: Lessons from the Past. World Bank Policy Research Paper, Volum 5016. Kalemli-Ozcan, S., Papaioannou, E. & Perri, F., 2012. Global banks and crisis transmission. National Bureau of Economic Research WP, 18209(18209).

[4] Kehoe, T. & Ruhl, V., 2008. Are shocks to the terms of trade shocks to productivity?. Review of Economic Dynamics, Volum 11, pp. 808-819. Kose, M. A., Otrok, C. & Whiteman, C. H., 2003. International Business Cycles: World, Region, and Country-Specific Factors. The American Economic Review, 93(4), pp. pp. 1216-1239.

[5]Kose, M. A., S., P. E. &Terrones, M. E., 2006. How do trade and financial integration affect the relationship between growth and volatility?. Journal of International Economics, Volum 69, pp. 176-202.

[6] Levine, R., 2005. Finance and growth: Theory and evidence. Handbook of economic growth, Volum 1, pp. 865-934. Love, I., Preve, L. &Sarria-Allende, V., 2007. {Trade credit and bank credit: evidence from recent financial crises}. Journal of Financial Economics, 83(2), pp. 453-469. Malouche, M., 2009. Trade and trade finance developments in 14 developing countries post September 2008-a World Bank survey. World Bank Policy Research WP, 5138(5138).