# Cosine and Soft Cosine Similarity-Based Anti-Phishing Model

**Chapter** · March 2021

DOI: 10.1007/978-981-33-4676-5_12



**3 authors**, including:

Parvinder Singh
Deenbandhu Chhotu Ram University of Science and Technology, Murthal
**58** PUBLICATIONS **165** CITATIONS
SEE PROFILE

Jasvinder Kaur
PDM College of Engineering
**7** PUBLICATIONS **0** CITATIONS
SEE PROFILE

# Chapter 12
# Cosine and Soft Cosine Similarity-Based Anti-Phishing Model


**Bhawna Sharma, Parvinder Singh, and Jasvinder Kaur**

**Abstract** Phishing attack has posed a greater threat to user information over network. In addition to the existence of various disguise illegal URL's, instances had been seen when users are redirected to phishing URL's that challenges their privacy concern. In the current work, author tried to develop an effective anti-phishing method based on hybrid similarity approach combining cosine and soft cosine similarity that measures the resemblance between user query and database. The strength of the proposed hybrid approach is further enhanced with the incorporation of feed forward backpropagation neural network (FFBPNN) so as to validate the similarity-based predictions. The model evaluated against 3000 sample files demonstrated to effectively detect phishing attacks with positive predictive value, true positive rate and F-measure of 71.9%, 72.6% and 72.23%, respectively.

## 12.1 Introduction

Modern advances in the technological sector have raised the popularity of Internet technology. Presently, none of the field exists that remains untouched by the network frame work of Internet. The rising popularity social media including Twitter, Instagram and Facebook further adds up to the popularity of Internet technology as an indispensable daily need [1]. It is observed that the number of people using social media has been nearly doubled since 2014. As shown in Fig. 12.1, 9% rise in the

B. Sharma · P. Singh (✉)
Department of Computer Science and Engineering, Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Sonepat, Haryana, India
e-mail: parvindersingh.cse@dcrustm.org

B. Sharma
e-mail: bhawnash024@gmail.com

J. Kaur
Computer Science and Engineering, PDM University, Bahadurgarh, Haryana, India
e-mail: jasvinder.kaur@pdm.ac.in



R. Kountchev et al. (eds.), *New Approaches for Multidimensional Signal Processing*, Smart Innovation, Systems and Technologies 216,
https://doi.org/10.1007/978-981-33-4676-5_12

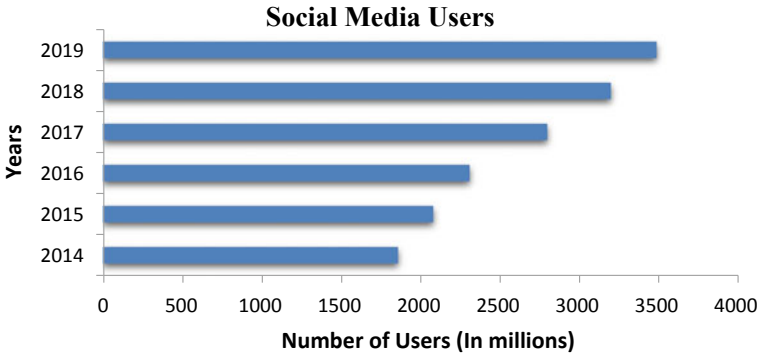


**Fig. 12.1** Rising popularity of social media

number of users has been observed in last year [2]. Such an enormous rise in the network traffic has significantly raised twofold challenges. One is malicious attacks and other is requirement of advanced hardware. This had further led to the discovery of big data computing technologies with more sophisticated hardware configuration. Internet has become a means that connects numerous users sharing information in the form of media, images, videos, files, etc., that also require attention for the risk of privacy leakage in one way or the other.

Phishing is one of the cyber-attacks that frequently appear in personal computers and mobile platforms. It is a criminal mechanism taking advantage of both social engineering and technical subterfuge that advertises and sends illegal links to users to deception their private information and financial credentials. Spoofed emails are employed as legitimate business tricking recipients to accidentally share their login and password details. In other words, phishing is defined as stealing someone private information by befooling them to be genuine [3–5]. Majority of the Internet users get fascinated by these illusions and get trapped. Software-as-a-Service (SaaS) and webmail services are the industry sectors, which are the main targets of phishing [6].

Researchers around the world have been constantly involved to tackle phishing attacks and deploy various anti-phishing protocols. It is found that one-third of the anti-phishing mechanisms are rule-based prevention methods implemented in recent past [7]. The major limitation adjoining this mechanism is the adaptability. This means that there is requirement of incorporation of new data of rule set with every new phishing attack. To deal with adaptability issues, authors have used swarm intelligence with machine learning [8]. Some similarity-based approaches have also been proposed that compares the phishing with legitimate websites [9]. In the current research, authors used rule-based architecture to develop an anti-phishing protocol based on the cosine and soft cosine similarity index powered by the machine learning design. Cosine similarity measures the resemblance between two documents independent of the document size. In Fig. 12.2, cosine angle is reflected by a set of vectors representing query and repository vectors projected in the space. The plus point of this similarity aspect is the fact that larger documents can still be expected

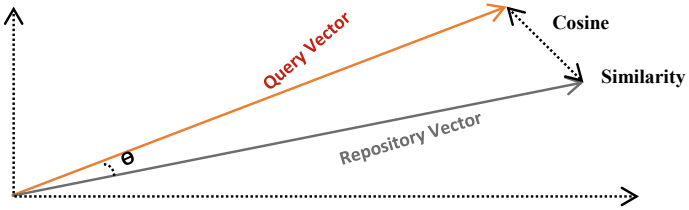


**Fig. 12.2** Cosine similarity

to get oriented in the environs. Thus, smaller cosine angle reflects higher similarity, for instance, cosine 0º reflects complete similarity as 1. More detailed mechanism is discussed in the later part of the article.

Author had introduced phishing attacks and related statistics in the current section. Section 12.2 summarizes the research revolving around anti-phishing protocols deployed by researchers, and Sect. 12.3 describes the proposed methodology based on cosine and soft cosine followed by Sect. 12.4 dedicated for observations and discusses the accomplish result. The paper is concluded in Sect. 12.5.

## 12.2 Literature Review

This section covers various types of protocols proposed to deal and prevent phishing attacks. Ramanathan et al. in [7] had proposed PhishGILLNET as multi-layered anti-phishing model. The technique demonstrated effective results with a prerequisite that webpage should be in HTML and MIME formats [7]. Li et al. [10] proposed a novel anti-phishing method based on ball-support vector machine (BVM) to distinguish a malicious URL from a genuine. In the process they extracted various topological features of the website and analysis, 12 out of them followed by the BVM-based vector analysis. Evaluation against SVM proved BVM to be highly effective in detecting phishing websites with a relatively slower speed for big data [10]. Kaur and Kalra [11] had proposed a five-tier anti-phishing design to protect. The hybrid approach analyses the URL and reflects the page status as secure or phishing website. [11]. In 2016, Nguyen et al. proposed a novel neuro-fuzzy ideal for detecting phishing attacks. The technique had employed a dataset of legitimate (10,000) and phishing websites (11,660) that was trained using neural network with adaptive learning rates. The results of the study showed its effectiveness in identifying the phishing websites [12]. Sonowal and Kuppusamy [13] developed PhiDMA as a multi-layered anti-phishing architecture divided into five layers corresponding to whitelist, URL feature, signature, string matching and score layer. The model was developed to offer easy access to even visually impaired individuals with 92.72% phishing detection accuracy [13]. Ugochi [14] focused his research towards depth analysis of IP address and URL cosine similarity in order to identify phishing URLs.

Experimental evaluation proved to be highly effective against 100 phishing URL used in the study with least memory requirement [14]. Makki et al. [15] postulated a cost-sensitive K-nearest neighbour (KNN) approach enhanced with cosine similarity to identify cheat instances affecting financial market, money transactions, telecommunication and credit card. The model proved to outperform the approach based on traditional KNN alone [15]. Jain et al. in 2018 had presented an innovative approach that was based on outstanding hyperlink features that aids in the recognitions of phishing attacks. It offered an effective client side solution and on logistic regression classifiers and it exhibited an accuracy of even higher than 98.4% [16]. Krodestani and Shajari in [9] proposed a novel textual similarity-based method to identify phishing sites. The method was evaluated against real website and demonstrated optimal phishing detection accuracy while discriminating between phishing and legitimate website and guides user towards genuine website [9]. Azeez in [17] had developed PhishDetect technique aimed to identify phishing websites by evaluating URL features and web contents. The technique proved to be highly efficient in identifying the phishing URLs [17]. Morovati et al. [18] dedicated their research study towards the identification of phishing attacks spread through phishing emails. To achieve distinguishable results, they had incorporated email forensic analysis with machine learning methodology [18]. In the same year, Zhu et al. had proposed OFS-NN as a phishing website detector. This model was based on optimal feature selection approach followed by neural network (NN) technique. They had developed an optimal classifier that could accurately detect different types of phishing websites [19]. Li in [20] had developed an architecture that mediates a cascade of Kaizen events. In this approach, the cosine similarity of data objects is used to classify protection levels. Experimental evaluation had shown that the designed fuzzy architecture proved to be competent as compared to other methods to support Kaizen architecture to identify susceptibility of web applications [20].

## 12.3 Proposed Methodology

### *12.3.1 Dataset*

Current research work employs the dataset obtained from *PhishTank* [21]. The downloadable data attributes consist of Phishing Id, URL information, type of target, online status, etc. It also presents a community-based evaluation platform where users query is classified to be phishing or legitimate-based votes. The site could be accessed at https://www.phishtank.com/.

The proposed anti-phishing design is largely a two-layered architecture. First layer is calculating the website similarity based on hybrid cosine and soft cosine between the input user query and the database repository. The second layer functions as a validation layer to check the effectiveness of the prediction performed by the first layer. This layer classifies the phishing and non-phishing sites based on a multiclass

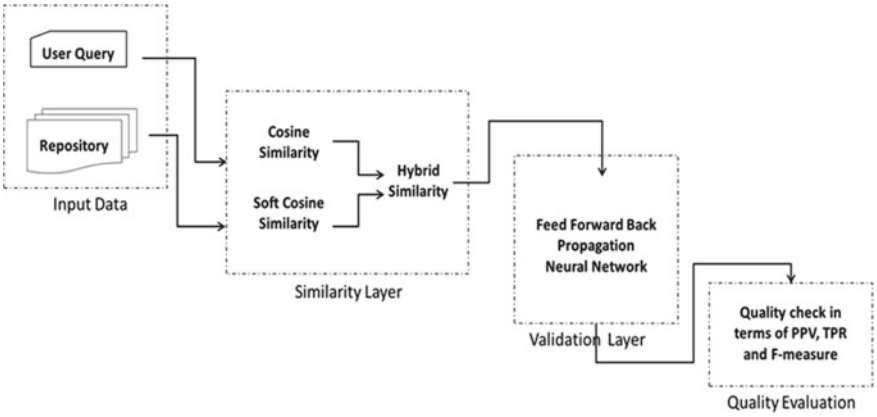


**Fig. 12.3** Proposed multi-layered anti-phishing architecture

neural network (NN). The quality of the proposed architecture is evaluated in terms of positive predictive value (PPV), true positive rate (TPR) and F-measure. The overview of the steps is shown in Fig. 12.3.

### *12.3.2 Similarity Layer (SL)*

The layer is dedicated to perform similarity predictions between the query or the test data and the repository. To achieve this, author had proposed a hybrid cosine and soft cosine-based similarity predictions that take advantage of angular co-relation established between the query and the repository vector as shown in Fig. 12.3, where "ɵ" defines the angle stretched by the two vectors. Mathematically, similarity is calculated as follows:

$$\text{Cosine}_{\text{Sim}} = \sum_{i=1}^{n} \frac{Q_{\text{Vect}(i)} R_{\text{Vect}}}{\sum_{i=1}^{n} (Q_{\text{Vect}})^2 \sum_{j=1}^{n} (R_{\text{Vect}})^2} \tag{12.1}$$

where, $\text{Cosine}_{\text{Sim}}$ represents, cosine similarly observed between user query represented by $Q_{\text{Vect}}$ and repository represented by $R_{\text{Vect}}$. The pseudocode to compute the similarity is summarized in Algorithm 1.

**Algorithm 1: Pseudocode for Cosine Similarity.**

1. Input: $R_{\text{Value}}$ // Repository data values
2. For each $I_{\text{Val}} \text{in} R_{\text{Value}}$ // Scan Every data value present in the repository
3. Vectorization of repository data:
   $R_{\text{Vect}} =$ conversion($R_{\text{Value}}$) // prune out stop words from the list.

4. Isolate words from repository data
   $w_{\text{data}} = \text{Segregate}(R_{\text{Vect}})$//segregate words from repository files.
5. Eliminate Stop Words
   $w_{\text{stop}} = \text{Remove}(w_{\text{data}})$ //removal of stop words.
6. ASCII code generation
   $w_{\text{data}_{\text{ASCII}}} = \text{ASCII}\left(w_{\text{stop}}\right)$ // generate ASCII Code for each word.
7. Calculate Cosine Similarity:

$$\text{Cosine}_{\text{Sim}} = \sum_{i=1}^{n} \frac{Q_{\text{Vect}(i)} R_{\text{Vect}}}{\sum_{i=1}^{n} (Q_{\text{Vect}})^2 \sum_{j=1}^{n} (R_{\text{Vect}})^2}$$

   Store to List
9. Output: $\text{Cosine}_{\text{Sim}}$ //Cosine similarly between query and repository.
10. End for

The above algorithm calculates the cosine similarity to predict the phishing sites or URLs. In the process, it first converts the data values to vectors and evaluates the stop words present in the repository data and the query data. Another similarity aspect, i.e., soft cosine is also calculated that takes the advantage of the same algorithmic flow except the similarity calculation made in step 7. The mathematically soft cosine similarity is calculated as follows:

$$\text{SCosine}_{\text{Sim}} = \frac{\sum_{i,j}^{n} Q_{\text{Vect}_i} R_{\text{Vect}_j}}{\sum_{i,j}^{n} Q_{\text{Vect}_i} Q_{\text{Vect}_j} \sum_{i,j}^{n} R_{\text{Vect}_i} R_{\text{Vect}_j}} \tag{12.2}$$

where, $\text{SCosine}_{\text{Sim}}$ represents the soft cosine similarity observed between user query represented by $Q_{\text{Vect}}$ and the repository data represented by $R_{\text{Vect}}$.

Further, the hybrid similarity prediction is performed by combining the observed similarity predicted by individual similarity calculations, i.e., using cosine similarity and soft cosine similarity as follows:

$$H_{\text{Sim}} = \text{Cosine}_{\text{Sim}} + \text{SCosine}_{\text{Sim}} \tag{12.3}$$

where hybrid similarity prediction is represented by $H_{\text{Sim}}$ based on the similarity predictions made by cosine and soft cosine similarity calculators represented by $\text{Cosine}_{\text{Sim}}$ and $\text{SCosine}_{\text{Sim}}$, respectively. The similarity calculations made in this layer is sent to validation layer.

### *12.3.3 Validation Layer (VL)*

The current layer evaluates the similarity predications of the similarity layer based on neural network (NN). It consists of input layer that inputs the user query, hidden

layer that act as a processing framework based on the weights and the output layer that returns the classification results. It is important to understand that raw data from input layer is passed to the hidden layer in the form of sigmoid function. Algorithm 2 summarizes the steps employed in the validation of the similarity predictions using neural network.

**Algorithm 2: Pseudocode for Validation using NN.**

1. Input: $H_{\text{Sim}}$ // hybrid similarity value
2. Initialize Variables:
   $T_{\text{Val}}$ // training value.
   $G_{\text{Val}}$ // group value.
3. Assign training value:
   $T_{\text{Val}} = H_{\text{Sim}}$ // assign respective training value.
4. Assign group value:
   $G_{\text{Val}} = G_{\text{num}}$ // group value is represented by respective group number.
5. Assign NN parameters:
   $D_{\text{Ratio}} = 0.7$ // distribution ratio.
   $CV_{\text{ratio}} = 0.15$// cross validation ratio.
   $T_{\text{ratio}} = 0.15$ // test ratio.
   $N_{\text{num}} = 20$ neurons //number of neurons.
6. Initialize Neural Network for Training
   $\text{Train}_{\text{NN}}(T_{\text{Val}}, G_{\text{Val}}, N_{\text{num}})$ // initialize Neural Network.
7. Start Training Neural Network
8. For each $X$in $F_{\text{classified}}$ //for every value in classified frame
9. If $F_{\text{Val}} == T_{\text{Val}}$ // classified result matches with the training value
10. $T_{\text{class}} + +$ //auto increment True classified class label
11. Else
12. $F_{\text{class}} + +$ //auto increment False classified class label
    End if
    End for

The similarity predications made in the previous layer are arranged in groups. The blocks in a group represent the similarity calculations corresponding to each repository. However, similarity predictions corresponding to a repository are represented by a single group tagged by repository name. The obtained sets then undergo supervised learning using neural network (NN). This multiclass classifier is used to distinguish phishing sites from legitimate sites and URLs. The group value and classified value are compared for each query value. If the group value and the classified values match, NN classifies the query to the true else false.

## 12.4 Result and Discussions

The phishing predications made by the proposed framework are evaluated in terms of demonstrated quality parameters, namely positive predictive value (precision), true positive rate (recall) and F-measure.

a. Positive predictive value (PPV) is represented by the number of true detections made by the prediction model in comparison with the total number of detections. Mathematically, it can be calculated as follows:

$$\text{PPV} = \frac{T_{\text{positive}}}{\left(T_{\text{positive}} + F_{\text{positive}}\right)} \tag{12.4}$$

where $T_{\text{positive}}$ and $F_{\text{positive}}$ represent true positive and false positive detections, respectively.

b. True positive rate (TPR) defines a number of positive results obtained that are actually correct or positive. It can be calculated as follows:

$$\text{TPR} = \frac{T_{\text{positive}}}{\left(T_{\text{positive}} + F_{\text{negative}}\right)} \tag{12.5}$$

where $F_{\text{negative}}$ represents the false negative detections.

c. F-measure represents the harmonic mean of above two parameters. It is calculated by product and arithmetic summation as follows:

$$F_{\text{measure}} = 2 * \left(\frac{\text{PPV} * \text{TPR}}{\text{PPV} + \text{TPR}}\right) \tag{12.6}$$

The values obtained using above relationships are summarized in Table 12.1. The column 2, 3 and 4 represents the average value of PPV, TPR and F-measure observed for respective number of files mentioned in column 1.

It is generalized from Table 12.1 that increase in the number of test files increases the average value of each considered parameter. Figure 12.4 shows the average value of PPV, TPR and *F*-measure for different number of test files. It concludes that

**Table 12.1** Average value of PPV, TPR and *F*-measure

| Number of test files | PPV | TPR | *F*-measure |
|---|---|---|---|
| 100 | 0.709 | 0.718 | 0.713 |
| 200 | 0.714 | 0.721 | 0.717 |
| 500 | 0.715 | 0.722 | 0.718 |
| 1000 | 0.719 | 0.725 | 0.722 |
| 2000 | 0.725 | 0.729 | 0.727 |
| 3000 | 0.732 | 0.738 | 0.735 |

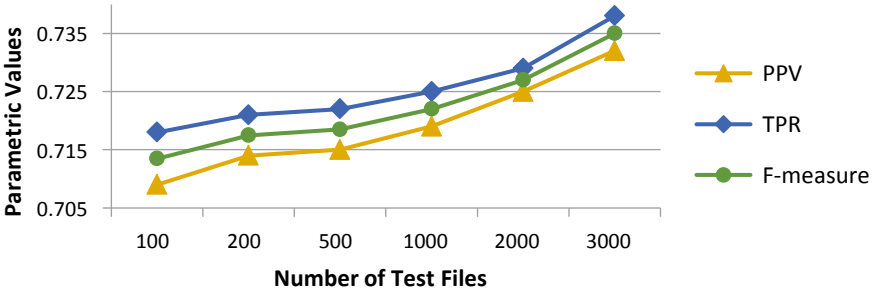


**Fig. 12.4** Average value of PPV, TPR and *F*-measure

best performance (maximum value of each parameter) is obtained by considering maximum number of test files.

## 12.5 Conclusion

In the current work, author has proposed an anti-phishing framework based on new rule-based architecture. The first layer of anti-phishing model performs phishing detection based on cosine and soft cosine similarity followed by neural network machine learning for performing cross-validation of the prediction results. The quality of results is evaluated in terms of PPV, TPR and F-measure. It is observed that the prediction model showed average enhanced PPV of 2.3%, TPR of 2% and *F*-measure of 2.15% over 3000 test files. However, an average value of PPV of 0.719, TPR of 0.726 and *F*-measure of 0.722 is observed that proved the effectiveness of the proposed anti-phishing design.

**Acknowledgements** This work is part of bilateral Indian-Bulgarian cooperation research project between Technical University of Sofia, Bulgaria and Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Sonepat, India under the title "Contemporary Approaches for Processing and Analysis of Multidimensional Signals in Telecommunications", financed by the Department of Science and Technology (DST), India and the Ministry of Education and Science, Bulgaria.

## References

1. Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L. F., Hong, J., Nunge, E.: Anti-phishing Phil: the design and evaluation of a game that teaches people not to fall for phish. In: Proceedings of the 3rd Symposium on Usable Privacy and Security, pp. 88–99. ACM, Pittsburgh, Pennsylvania, USA (2007)
2. Published in Global Digital Report 2019. https://datareportal.com/. Last accessed 2020/02/21

3. Cao, Y., Han, W., Le, Y.: Anti-phishing based on automated individual white-list. In: Proceedings of the 4th ACM workshop on Digital identity management, pp. 51–6 0. ACM (2008)
4. Aksu, D., Turgut, Z., Üstebay, S., Aydin, M.A.: Phishing analysis of websites using classification techniques. In: International Telecommunications Conference, pp. 251–258. Springer, Singapore (2019)
5. Lam, T., Kettani, H.: PhAttApp: A phishing attack detection application. In: Proceedings of the 2019 3rd International Conference on Information System and Data Mining, pp. 154–158. Houston, TX, USA (2019)
6. Anti-Phishing Trend Report (Q2 2019). https://apwg.org/trendsreports/
7. Ramanathan, V., Wechsler, H.: phishGILLNET—phishing detection methodology using probabilistic latent semantic analysis. AdaBoost, and co-training. EURASIP J. Inf. Secur. **1**(1), 1–22 (2012)
8. Mensah, P., Blanc, G., Okada, K., Miyamoto, D., Kadobayashi, Y.: AJNA: Anti-phishing JS-based visual analysis, to mitigate users' excessive trust in SSL/TLS. In: 2015 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS), pp. 74–84. IEEE, Kyoto, Japan (2015)
9. Kordestani, H., Shajari, M.: A similarity-based framework for detecting phishing websites. Int. J. Adv. Res. Comput. Sci. **9**(1), 792–796 (2018)
10. Li, Y., Yang, L., Ding, J.: A minimum enclosing ball-based support vector machine approach for detection of phishing websites. Optik **127**(1), 345–351 (2016)
11. Kaur, D., Kalra, S.: Five-tier barrier anti-phishing scheme using hybrid approach. Inf. Secur. J. Glob. Perspect. **25**(4–6), 247–260 (2016)
12. Nguyen, L.A.T., Nguyen, H.K., To, B.L.: An efficient approach based on neuro-fuzzy for phishing detection. J. Autom. Control Eng. **4**(2), 159–165 (2016)
13. Sonowal, G., Kuppusamy, K.S.: PhiDMA–a phishing detection model with multi-filter approach. J. King Saud Univ. Comput. Inf. Sci. **32**(1), 99–112 (2020)
14. Ugochi, O. C.: A novel web page anti-phishing approach using URL cosine similarity and IP address comparison. In: International Conferences on WWW/Internet, ICWI 2018 and Applied Computing pp. 321–328. IADIS Press (2018)
15. Makki, S., Haque, R., Taher, Y., Assaghir, Z., Hacid, M. S., Zeineddine, H.: A cost-sensitive cosine similarity K-nearest neighbor for credit card fraud detection. Big Data Cyber-Secur. Intell. 42–47 (2018)
16. Jain, A.K., Gupta, B.B.: A machine learning based approach for phishing detection using hyperlinks information. J. Ambient Intell. Human. Comput. **10**(5), 2015–2028 (2019)
17. Azeez, N., Salaudeen, B., Misra, S., Damaševičius, R., Maskeliūnas, R.: Identifying phishing attacks in communication networks using URL consistency features. Int. J. Electron. Secur. Digit. Forensics **12**(2), 200–213 (2020)
18. Morovati, K., Kadam, S.S.: Detection of phishing emails with email forensic analysis and machine learning techniques. Int. J. Cyber-Secur. Digit. Forensics **8**(2), 98–108 (2019)
19. Zhu, E., Chen, Y., Ye, C., Li, X., Liu, F.: OFS-NN: an effective phishing websites detection model based on optimal feature selection and neural network. IEEE Access **7**, 73271–73284 (2019)
20. Lin, K.S.: New attack potential measurement method to kaizen event for web application security vulnerabilities. Int. J. Electron. Commerce Stud. **10**(2), 89–112 (2019)
21. PhishTank. https://www.phishtank.com/, last accessed 2020/02/01