Journal of Network and Systems Management (2024) 32:8
https://doi.org/10.1007/s10922-023-09784-7

A CNN-Based SIA Screenshot Method to Visually Identify Phishing Websites

Dong-Jie Liu[1] · Jong-Hyouk Lee[2]

Received: 30 June 2022 / Revised: 12 February 2023 / Accepted: 19 October 2023 /
Published online: 21 November 2023

Abstract

Phishing evolves rapidly nowadays, causing much damage to finance, brand reputation, and privacy. Various phishing detection methods have been proposed along with the rise of phishing, but there are still research issues. Phishing websites mainly steal users' information through visual deception and deep learning methods have been proved very effective in computer vision applications but there is a lack in the research on visual analysis using deep learning algorithms. Moreover, most research use balanced datasets, which is not the case in a real Web environment. Therefore, this paper proposes a security indicator area (SIA) which contains most security indicators that are designed to help users identify phishing sites. The proposed method then takes screenshots of SIA and uses a convolutional neural network (CNN) as a classifier. To prove the efficiency of the proposed method, this paper carries out several comparative experiments on an unbalanced dataset with much fewer phishing sites, which increases detection difficulty but also makes the detection closer to reality. The results show that the proposed method achieves the highest F1-score among the compared methods, while providing advantages on detection efficiency and data expansibility in phishing detection.

Keywords Phishing website detection · Security indicator area screenshot · Convolutional neural network · Deep learning

Dong-Jie Liu and Jong-Hyouk Lee have contributed equally to this work.

✉ Jong-Hyouk Lee
jonghyouk@sejong.ac.kr

Dong-Jie Liu
djliu@jnu.edu.cn

[1] College of Cyber Security, Jinan University, Guangzhou, China

[2] Sejong University, Seoul, Republic of Korea

1 Introduction

Phishing is a combination of "Fishing" and "Phone". Today's phishing attacks, however, exploit more different ways such as emails, websites, short messages, social media, etc. to carry out fraudulent activities. Phishers often disguise themselves as trusted brands and victims reveal their personal information and financial data to them. Nowadays, with the fast development of Internet of Things (IoT) with 5G-Advanced to meet the needs of the future ubiquitous smart society, much more terminals are connected to the Internet [1], which results in more targets of phishing.

One damage caused by phishing is financial loss. For instance, it was stated by Ponemon Institute that there was 3.77 million dollars loss caused by phishing only for the average-sized organization in their sample [2]. However, the financial losses of the organization may be recouped after a while, but the loss of brand reputation may take years to recover. In the event of an attack on the organization, the customer may suspend business with the organization in the future. In a Ponemon survey, 31% of respondents said they would break off a partnership if they were told about a data security breach. They also said they would immediately suspend contracts if third-party suppliers suffered such incidents [3]. Another possible loss is privacy. For instance, if the account or key of an indoor security camera, one typical IoT product, is obtained by phishers, it is not hard to imagine the terrible consequences of this privacy leak [4].

Luckily, although the targets and channels of phishing are diversified, the final step of most phishing is the same: a phishing website[5, 6]. Therefore, phishing website detection is the main battlefield of anti-phishing [7–9]. To combat phishing, various anti-phishing measures are taken and improved nowadays. The most common measure is phishing detection and now most phishing detection methods are based on machine learning [10–13, 38]. However, there are still problems in machine learning based methods, which are summarized as follows:

- Most research extracts features from URLs or source codes of webpages and do not consider the visual fraudulence of phishing. Models based on text features cannot effectively recognize phishing websites that use redirection, hidden spam technologies or picture-in-picture techniques, and they can also be easily influenced by various languages on phishing sites and phishers can change the sites (e.g. URL) at a very low cost.
- With the development and popularization of deep learning methods, especially with its success in computer vision applications, there is still a lack of study in the application of deep learning algorithms in visual analysis of phishing websites.
- In most studies, there is more phishing website data than legitimate one, or the amount of data is balanced, which may help the classifier to achieve better performance more easily, but is different from the reality in the real Web environment where there are far more legitimate websites than phishing websites and would make the classification much more difficult.

Therefore, considering that more spam techniques like picture-in-picture are used by phishing websites and to eliminate the influence of various website languages and the easy change of texts, this paper proposes a detection method based only on images. This paper first proposes a Security Indicator Area (SIA), which contains most security indicators that are made to help users identify phishing sites, and so it is not a random capture of an image but interpretable and understandable; besides, it is light and saves computing resources and storage. Then it takes screenshots of SIA and uses deep learning algorithms, which requires no artificial features and thus is labor-saving and practical. It then proves the efficiency of this method by several comparative experiments on a constructed unbalanced dataset with much fewer phishing websites. The contributions of this paper are briefly summarized as follows:

- This paper analyzes visual counterfeiting of phishing websites, and proposes SIA as the input. Image input can help strenthen the detection of phishing websites that use spam techniques like picture-in-picture, eliminate the influence of various website languages worldwide and avoid the problem caused by easy change of texts on websites, which in other words, it is cross-language and practical. What is more, SIA contains most security indicators that are made to help users identify phishing sites, and so it is not a random capture of an image but interpretable and understandable; and at the same time it is small and automatically obtained so it saves computing resources, storage and labor.
- Instead of traditional machine learning methods, this paper uses deep learning algoritms in phishing detection and makes full use of CNN's powerful function in image classification and improves the accuracy and comprehensiveness of the detection model.
- Considering the reality in the real Web environment where there are far more legitimate websites than phishing websites, different from most research, this paper constructs an unblanced dataset and carries out several comparative experiments with related methods and research. The results prove the better performance of the proposed method well.

The organization of the rest paper is: Sect. 2 discusses related work on phishing detection methods; Sect. 3 describes the proposed method; Sect. 4 presents the comparative experiments and results and at last Sect. 5 summarizes this paper and presents possible future work.

2 Related Work

To combat phishing, many phishing detection methods have been developed, which can be categorized as machine learning based methods and non-machine learning based methods. Methods such as heuristics, blacklisting, visual similarity technique are typical non-machine learning based methods [14, 15]. Machine learning methods, including both classic machine learning and deep learning methods,have been used in many different research topics related to security, such as in intelligent acoustic system security [16–18] and attack detection [19, 20]. Similarly, in phishing website detection, machine learning methods are also the most popular methods. Algorithms such as AdaBoost, Naive Bayes, Random Forests (RF) and Support Vector Machines (SVM) are commonly used by classic machine learning based methods [9, 10, 12, 21–24]. Algorithms such as Convolutional Neural Network (CNN) and Long-Short Term Memory (LSTM)are used by a few deep learning based methods in phishing detection [25–27].

In most classic machine learning methods, although different features are extracted, most of them rely heavily URL and HTML, which can be easily influenced by spam techniques, various website languages and the easy change of texts (e.g.URL) on phishing websites. Visual features reflecting phishing websites' important characteristic—visual counterfeiting, which can avoid those problems, are often ignored. In addition, there is more phishing data than legitimate website data or the datasets are balanced, which may help the classifier algorithm earn a better performance but is not the case in real Web environment, where the number of legitimate sites is much bigger than phishing ones and would make the classification much more difficult.

For instance, Moghimi and Varjani collected 1158 phishing and 549 legitimate websites and employed SVM to classify the webpages based on 8 features, such as number of dots in URL and page resource identity but with no visual features [9]. Sahingoz et.al built a dataset with 36,400 legitimate and 37,175 phishing URLs and proposed natural language processing based features. They adopted different machine learning algorithms and Random Forest algorithm with only NLP based features performed best [12]. Rao and Pais extracted three kinds of features, including URL obfuscation features, third-party-based and hyperlink-based features, still with visual features. They collected 2119 phishing and 1407 legitimate sites and used 8 algorithms such as Random Forest, SVM, AdaBoostM1 and logistic regression, among which Random Forest performed the best [10].Cuzzocrea et.al extracted features from URL, domain etc. and used algorithms such as J48 and Random Forest as the classifier and J48 got the highest F-measure as 91.9%, but they did not introduce their datasets clearly[28].

Only a few studies using classic machine learning based methods include visual features but they are usually limited to only one visual feature. Besides, similarly, the dataset covers more phishing sites than legitimate ones or the datasets are balanced. For instance, Jain and Gupta chose 20 features, only one feature is about favicon, an image icon of a website. They collected 2141 phishing and 1918 legitimate websites and RF outperformed the other common algorithms [11]. Chiew et.al extracted only logo image and used SVM as the classifier. The proposed method was carried out on a small dataset with 500 phishing and 500 legitimate websites [8]. Lokesh and BoreGowda used most features from URL, HTML and domain and they add favicon,which is good, and compared different algorithms like Random Forest, K nearest neighbours and Decision Tree; but the size of the data is not clearly described[39]. One exception is that Geng et.al used unbalanced datasets, but its problem is that it only used favicon and logo image features and others are text features and it only used C4.5 as the classifier [7], so it is unknown whether this algorithm is the best when compared with other algorithms, especially with deep learning methods.

Although it has been used in different fields since then, there is still a lack of research using deep learning algorithms in phishing website detection and the problems were similar to those using classic machine learning based methods. Most research do not makes visual analysis and some still use balanced datasets. For example, Chen et al.and Liang et al. both used only URL as the input and Long Short-Term Memory (LSTM) as the classifier. Besides, the datasets were balanced [26, 30]. Tajaddodianfar et.al also used URL as the input and chose CNN as the classification algorithm and the datasets were not introduced clearly [31]. Only Wei et al. used unbalanced datasets with more legitimate sites but they used CNN to analyze only URLs and the performance is not very satisfying with only 86.63% accuracy [25]. Alshehri et al. used CNN based on character level embedding to detect phishing URLs and the ratio of legitimate and phishing sites are approximately 2:1[40]. Dilhara compared different deep learning models such as CNN (1D), LSTM and GRU based only on URLs and opposite to reality, the phishing sites are even more than the legitimate ones[41]. To be noted, Hiransha et al. used CNN to detect phishing emails [27]. Although the target is not the same, it has reference value for phishing website detection, for using HTML files as the input should be also a possible way to detect phishing sites.

To present the discussion above more clearly, a brief summary of recent related works on phishing website detection is made (see Table 1).

From Table 1, it is quite obvious that only a very few studies using classic machine learning algorithms consider limited visual features and no deep learning based methods use visual as input, which is not scientific at all for phishing website detection, because phishing websites basically use visual counterfeiting to fool users into providing their private information. What is more, the datasets usually contain more phishing data than legitimate data or the datasets are balanced, which helps algorithms to achieve a seemingly good performance; but is not the real case in reality, where the number of legitimate websites is much larger than that of phishing sites and would make the classification much more difficult.

Therefore, this paper first proposes a Security Indicator Area (SIA), which contains most security indicators that can help users identify phishing sites.So it is not a random capture of an image but interpretable and understandable; besides, it is small and automatic and saves computing resources, storage and labor; and most importantly, it solves the problems caused by text features. This paper then takes screenshots of SIA and uses CNN as the classifier. Comparative experiments are conducted on unbalanced datasets with much fewer phishing sites and all the results prove the efficiency of the proposed method.

3 The Proposed Method

3.1 Security Indicator Area

From the works studied in Sect. 2, it is noticed that no matter what kind of machine learning methods researchers use, the text features they extract mainly come from URLs and source codes and visual features include favicon and logo image [8, 9, 11, 12, 25–27]. The disadvantages of text-based detection method

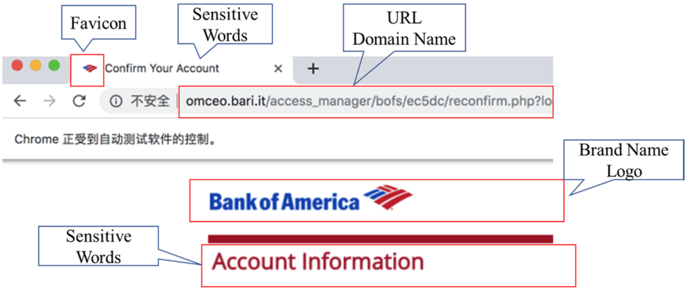
Table 1 A brief summary of recent related works on phishing detection

Researchers	Visual feature/input	Datasets	Algorithm
Moghimi and Varjani[9]	none	more phishing data	classic machine learning algorithm
Sahingoz et.al[12]	none	more phishing data	
Rao and Pais[10]	none	more phishing data	
Cuzzocrea et.al[28]	none	unclear	
Lokesh and BoreGowda[39]	favicon	unclear	
Jain and Gupta[11]	favicon	more phishing data	
Chiew et.al[8]	logo	balanced datasets	
Geng et.al [7]	favicon and logo	fewer phishing data	
Chen et al.[26]	none	balanced datasets	deep learning algorithm
Liang et al.[30]	none	balanced datasets	
Alshehri et al.[40]	none	fewer phishing data	
Wei et al. [25]	none	fewer phishing data	
Tajaddodianfar et al. [31]	none	unclear	
Dilhara[41]	none	more phishing data	
Hiransha et al.[27]	none	fewer phishing data	
Liu D.-J& Lee J.-H. (This paper)	SIA screenshots— containing most security indicators	much fewer phishing data	deep learning algorithm

are obvious. It has been emphasized in the first section that it can not solve the problem of the detection of phishing websites using spam techniques like picture-in-picture, nor can it solve the cross-language problem and low easy-change cost of phishing websites. What is more, about detection based on logo images, the problem is that legitimate websites such as promotional websites and sub-brand websites also have brand logos and it is easy to lead to misjudgment; also it is hard to locate the real logo since different website may have logos in different places. To solve such problems, it usually needs third-party search engine resources [8] which may easily reduce model efficiency and cannot be applied to real network environment.

Therefore, how to take advantage of these features and at the same time avoid their shortcomings is worth exploring. It is noticed that in URLs, number of dots, the use of IP address, https, domain names (which usually contain brand names) can all be used by users to identify phishing. For instance, PhishLabs identified phishing sites residing on more than 170,000 unique domains in 2017 and statistically 65.8% of all websites use “https”[32, 33]. In addition, besides favicon and logo image, the security indicators users can see in a website to identify phishing also include padlock icon and brand name.

After studying these common security indicators made to help users identify phishing, it is found that these features users can see directly in fact all locate in the top left quarter of the whole webpage. One example is provided in Fig 1. In our previous work, we took screenshots of the whole webpage and used the CNN algorithm, but it was for malicious website detection and the images were too large and took up too much storage space [34]. Therefore, since the security indicators mainly lie in the top left quarter and humanbeings can use them to identify phishing websites, taking screenshots of that area may also work out for computers. Considering that the area include most security indicators, it is called as SIA in this paper.



Chrome 正受到自动测试软件的控制。

Fig. 1 One example of SIA containing security indicators

3.2 The Proposed Approach

After studying some related approaches and classic models such as LeNet, AlexNet and GoogleNet [34, 38], we propose a model (see Fig. 2) and the process and model is described as follows:

1) The input layer is a resized screenshot. The original size of webpage screenshot is 1200*652 and it is resized as 256*256. The original size of SIA screenshot is much smaller as 600*250 and then is resized as 250*250. There are three channels so the final size of webpage screenshot is 256*256*3; and for SIA screenshot, it is 250*250*3.

2) There are three convolutional layers with 32 convolution kernels. The kernel size is 3*3. This paper adopts Rectified Linear Unit (ReLU) as the activation function presented in Eq. (1)and max pool with 2*2 filters.

$$ReLU(x) = \begin{cases} 0 & x \leq 0 \\ x & x > 0. \end{cases} \tag{1}$$

3) At last, it is the fully connected layer with 64 neurons. It uses a sigmoid function presented in Eq. (2), which is one of the most commonly used functions in

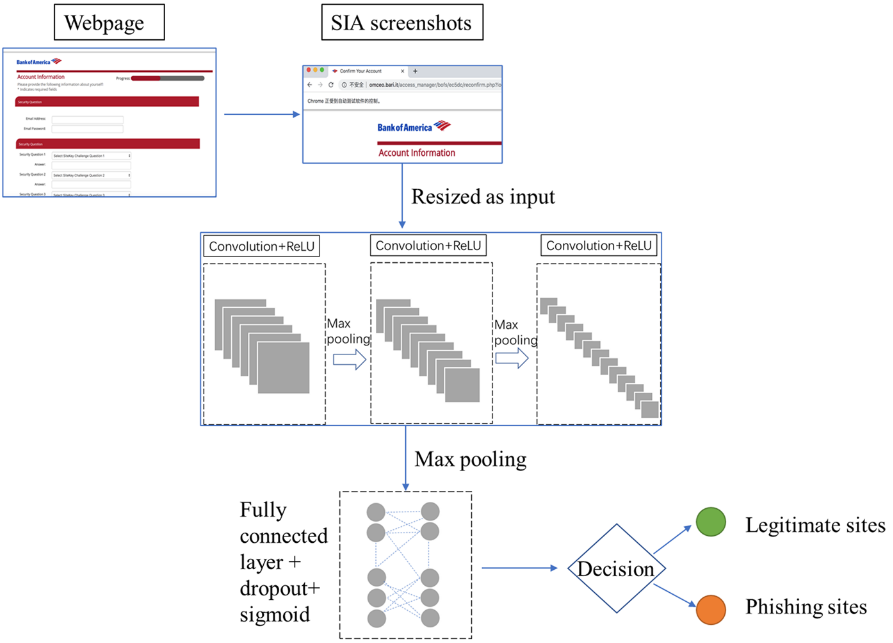


Fig. 2 The proposed approach

machine learning, especially for binary classification. Moreover, it is added a dropout layer to prevent overfitting.

$$sigmoid(x) = \frac{1}{1 + e^{-x}}. \tag{2}$$

TensorFlow is a famous numerical library dedicated to deep learning and Keras is a high-level API built on TensorFlow, Theano or CNTK, which is very easy to use[35, 36]. Therefore, this paper uses Keras in Python for CNN application and does not need to interact directly with the more complex TensorFlow. In addition, batch size is set as 32, rescale as 1/255, shear range and zoom range as 0.2, and horizontal flip is set true.

4 Evaluation

4.1 Definition of Metrics

In order to make the later comparative results clearer, the definitions of the adopted metrics in this paper need to be introduced. Before introducing the adopted metrics precision, recall and F1 measure, the basic concepts are first clarified.

As shown in Table 2, True Positive (TP) refers to the ratio of correct prediction on positive samples; False Positive (FP) refers to the ratio of wrong prediction on positive samples; True Negative (TN) refers to the ratio of correct prediction on negative samples; and False Negative (FN) refers to the ratio of wrong prediction on negative samples.

The adopted metrics in this paper are the precision, recall rate and F1-score, whose definitions are as follows:

- Precision is calculated as $\frac{TP}{TP+FP}$, which shows how many of the samples that are predicted to be positive are actually positive.
- Recall rate is calculated as $\frac{TP}{TP+FN}$, which shows how many of the positive examples in the whole dataset are predicted correctly.
- F1-score is calculated as $\frac{2*(Recall*Precision)}{Recall+Precision}$, which shows the overall performance of a method and is especially useful in unbalanced data categorization.

Table 2 Confusion Matrix

		Reality	
		positive (phishing)	negative (legitimate)
Predicted Label	positive (phishing)	True Positive (*TP*)	False Positive (*FP*)
	negative (legitimate)	False Negative (*FN*)	True Negative (*TN*)

4.2 Dataset Construction

So far, there is no public phishing dataset containing website screenshots, so we constructed a dataset including 3843 legitimate websites and 1593 phishing sites, with the screenshot, SIA screenshot, URL and HTML file of each site.

The URLs of legitimate and phishing websites are first needed. The legitimate URLs are obtained from websites like DMOZ and the phishing URLs are achieved from PhishTank. Both are obtained randomly. After removing the duplicated URLs, to make sure the websites are all active, their HTML files are checked to see if they are able to be downloaded. If the answer is yes, their URLs and HTML files are reserved otherwise the URLs are deleted. At last, each of the filtered URLs and HTML files is given an ID number. So far, we get the URLs and HTML files and the preparation before taking screenshots is ready.

Then what is needed to do is to open the web page through browsers just like what Web users do and take the picture of the webpage, which is exactly what users will see when they open the website. An API called WebDriver can just meet this need. WebDriver is based on Selenium for browser operation and it does not depend on any test framework or need to start other processes or install other programs other than the necessary browser drivers. It supports a variety of programming languages and this paper uses Java. WebDriver here works as a third-party library for Web automation in Java. Besides, it also supports multiple browsers and this paper uses Chrome.

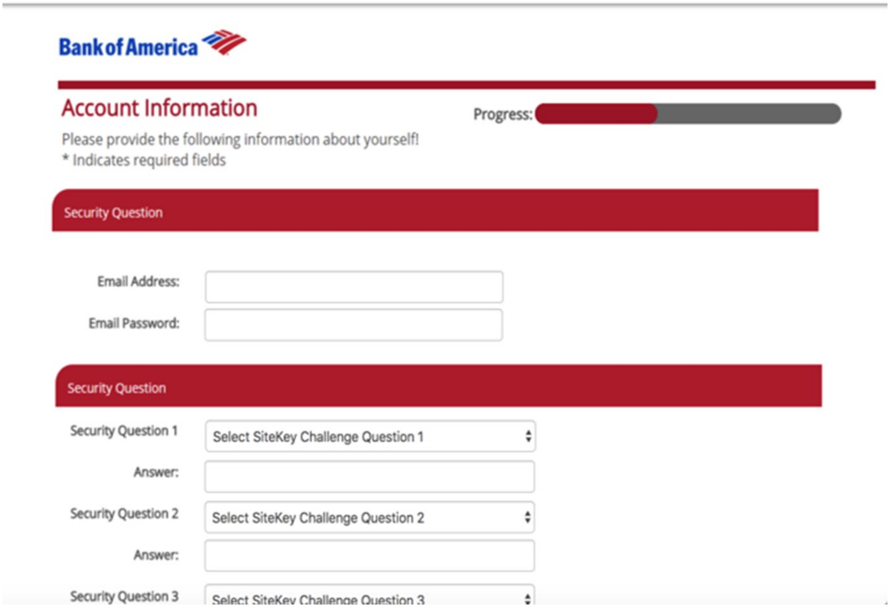
Two kinds of screenshots are taken, one is the webpage screenshot, the other is the SIA screenshot. The names of the screenshots are the same with the ID number of the URLs and HTML files, but the amount is smaller. Because to save time, if the time of opening a website is too long, the program will give up and move to the next one.

Figure 3 and Fig. 4 are the samples of the screenshots of one same phishing site. One is the webpage screenshot and the other is the SIA screenshot.

4.3 Comparative Experimental Results

The datasets of SIA screenshots are split into training, validation and testing sets with the ratio of 7:2:1. The epochs are set as 51 times at first, for in our previous work [34], which was about the detection of malicious websites with webpage screenshots on different and larger datasets, the model was overfitting after 51 times. Each epoch takes about 6 min in the training process and the performance of the model on training and validation sets are recorded as the epoch times increase. However, from the recorded performance, it is noticed that on the validation set of SIA screenshots, the general trend of F1-score is rising, though at a low speed and the loss on the validation set does not decrease at a certain epoch clearly. these make 51 epochs worth reconsidering.

Based on the discussion above, the epochs are then set as 80. This time, as shown in Fig. 5,the 68^{th} epoch is the turning point, because the loss on the validation set



Bank of America

Account Information

Progress:

Please provide the following information about yourself!
* Indicates required fields

Security Question

Email Address:

Email Password:

Security Question

Security Question 1 Select SiteKey Challenge Question 1

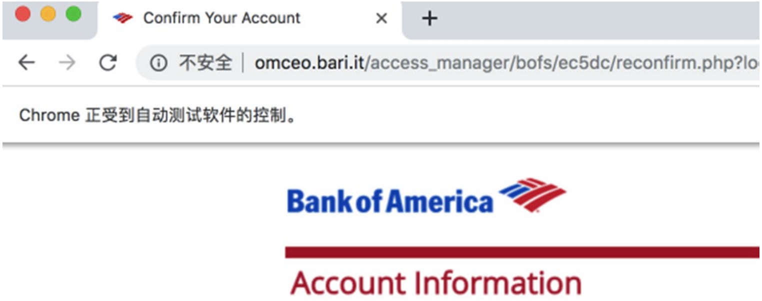
Answer:

Security Question 2 Select SiteKey Challenge Question 2

Answer:

Security Question 3 Select SiteKey Challenge Question 3

Fig. 3 Sample of webpage screenshot



Confirm Your Account

不安全 | omceo.bari.it/access_manager/bofs/ec5dc/reconfirm.php?lo

Chrome 正受到自动测试软件的控制。

Bank of America

Account Information

Fig. 4 Sample of SIA screenshot

starts to increase and F1-score on the validation set does not become higher after the certain epochs.

The results are presented in Table 3.

Then to further verify the effectiveness of the proposed method, this paper compares the proposed method with some current methods, including both classic machine learning methods and deep learning methods.

Comparative experiments with traditional machine learning methods As discussed before, it is noticed that most phishing detection methods use features extracted from URLs and HTMLs. What is more, SIA screenshots in fact contain most text information for users to identify phishing sites, like "https" in URLs

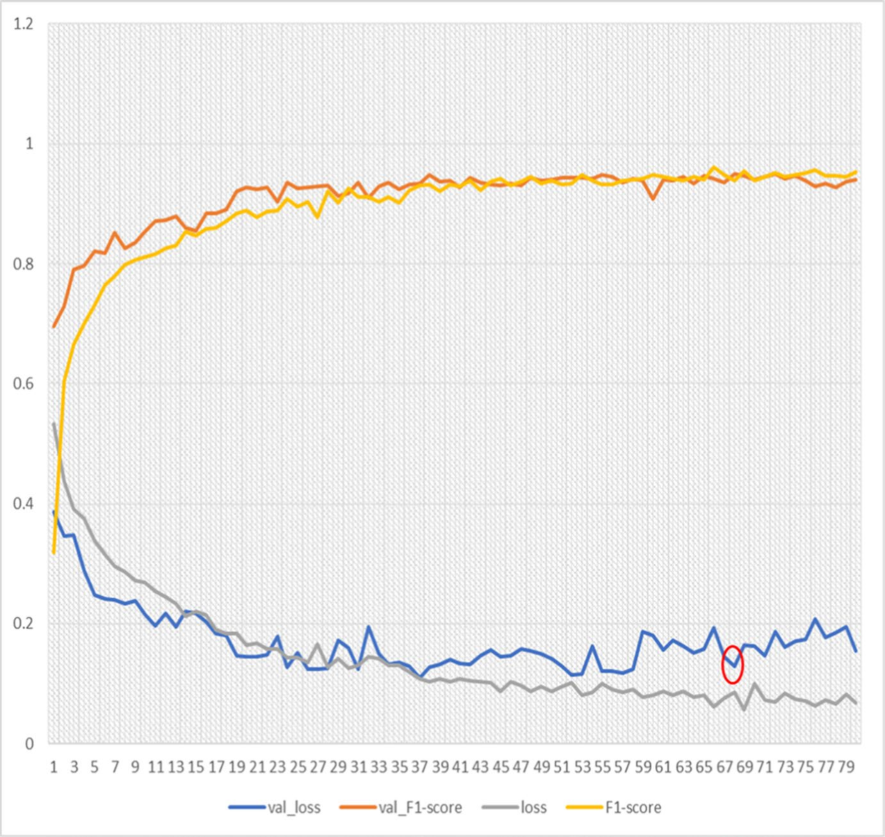


Fig. 5 The performance of CNN on SIA screenshots with 80 epochs

Table 3 Comparative results with traditional machine learning methods

Categorization Method	Precision	Recall	F1-score
The proposed method	0.981	0.944	0.962

and brand names in webpage titles. Therefore, this paper also extracts text features from URLs and HTMLs.

As has mentioned in Sect. 4.2, URLs and HTML files have been downloaded when the website is checked active, and the screenshot set is the subset of them. So after matching and removing the redundant URLs and HTML files, the URLs and HTML files are prepared and ready for the comparative experiments.

After the data is ready, 220 dimensional features metioned in different research are extracted from URL set and HTML set, preparing for the comparative experiments. There are 94 dimensional features extracted from URLs, including four categories:

- The first one is whether the URL contains IP address. Some phishing sites use IP address in their URLs, which will not take place in legitimate sites, so if a URL contains IP address, it is labeled as 1 (phishing sites), otherwise it is labeled as 0 (legitimate sites).
- Besides, legitimate websites, especially brand websites, all use “https” (Hyper Text Transfer Protocol) in their URLs, but it does not always happen in phishing sites. So “https” is the other feature extracted in URLs.
- In addition, the number of dots in hosts is another dimension. According to the statistics on our datasets, the average number of dots in hosts is 2.89 and if the number of dots is more than 5, it is more likely to be a phishing site.
- The last one is whether the URL contains brand string or brand domain name, such as “PayPal” and “Bank of America”.

From HTML, 126 dimensional features are extracted, including three categories:

- The first one is sensitive words feature, which covers 17 sensitive words, such as “register”, “sign in”, “log in”, “pay”, “bank”, “credit card” and etc..
- The second one is brand name feature, which has 104 brand names, such as Facebook and Amazon.
- The last one is HTML tag features, which has 5 dimensions, such as “refresh”, “submit” and action=“https”.

Different popular classic machine learning algorithms, including Naive Bayes, AdaBoost M1, C4.5 Decision Tree, Bagging, Random Forest, SMO (Sequential Minimal Optimization) with Polynomial Kernel are adopted and the results of the best four algorithms are presented in Table 4.

From Table 4, it is seen that among all the classic machine learning algorithms, Random Forest achieves the best results in all the three metrics. Its precision is almost the same with that of CNN on SIA screenshot, but the recall rate is much lower. It means that although it performs well in identifying phishing sites in all the predicted positive items, it is not good at finding out all the phishing sites among the whole dataset, which implies that it is not a good choice when put into real Web environments. This is probably because text features are easily influenced by languages and cannot deal with websites using spam techniques, while methods based

Table 4 Comparative results with traditional machine learning methods

Categorization Methods	Precision	Recall	F1-score
C4.5 Decision Tree	0.956	0.813	0.878
Bagging(C4.5)	0.938	0.850	0.892
SMO (Sequential Minimal Optimization) with Polynomial Kernel	0.958	0.850	0.901
Random Forest	0.973	0.894	0.932
The proposed method	0.981	0.944	0.962

on images can avoid these problems. F1-score is the calculated value based on precision and recall rate so it is very clear that CNN on SIA screenshot outperforms Random Forest on text features.

Comparative experiments with deep learning methods After proving the better performance of the proposed method than classic machine learning algorithms, it is also necessary to compare the proposed method with some current deep learning methods, including CNN[25, 27], Bi-LSTM[30], MLP [43], transformer[44] and CNN-BiLSTM[42] and the comparative results are presented in Table 5.

From Table 5, it is seen that the proposed CNN based SIA screenshot method still has the best performance. Apart from the proposed method, the method of CNN-BiLSTM based on URL [42] achieves the highest F1-score as 0.953. However, one thing that needs to be pointed out is that this method is only suitable for fixed datasets and are not applicable in real Web environment. Because phishing sites only need to make a slight change in their URLs and it can become very difficult to detect them only using URL as the input. What is more, also for methods based on HTML, they are easily impacted by various languages around the world since phishing is a global problem; and methods on texts cannot detect websites using spam techniques like picture-in-picture. This also helps explain why the proposed method performs better.

In addition, usually there is a negative correlation between precision and recall rate, so it is very hard to get high precision and recall rate at the same time; but deep learning algorithms show their robustness in both visual and text input. And when there is new data joining in, deep learning methods are very convenient to use, for although it takes some time when training the data, once the model is trained, it is very efficient to test new data and it does not need to extract possible new features manually.

Comparison with webpage screenshot Condering that our previous work detects malicious websites with webpage screenshots [34], this paper also compares the proposed method with webpage screenshots.

Based on the previous discussion, the epochs are also set as 80. As shown in Fig. 6, for webpage screenshot dataset, it is quite obvious that the 70th epoch is the turning point because the loss on the validation set starts to increase and F1-score on the validation set does not become higher after the certain epochs.

Table 5 Comparative results with deep learning methods

Method	Precision	Recall	F1-score
Wei B. et al. [25]	0.961	0.931	0.946
Liang Y. et al. [30]	0.955	0.938	0.946
Hiransha M. et al. [27]	0.939	0.963	0.951
Zhang Q. et al. [42]	0.966	0.940	0.953
Al-Ahmadi S. and Lasloum T. [43]	0.932	0.911	0.921
XU P. [44]	0.918	0.902	0.910
Bi-LSTM on HTML	0.950	0.944	0.947
The proposed method	0.981	0.944	0.962

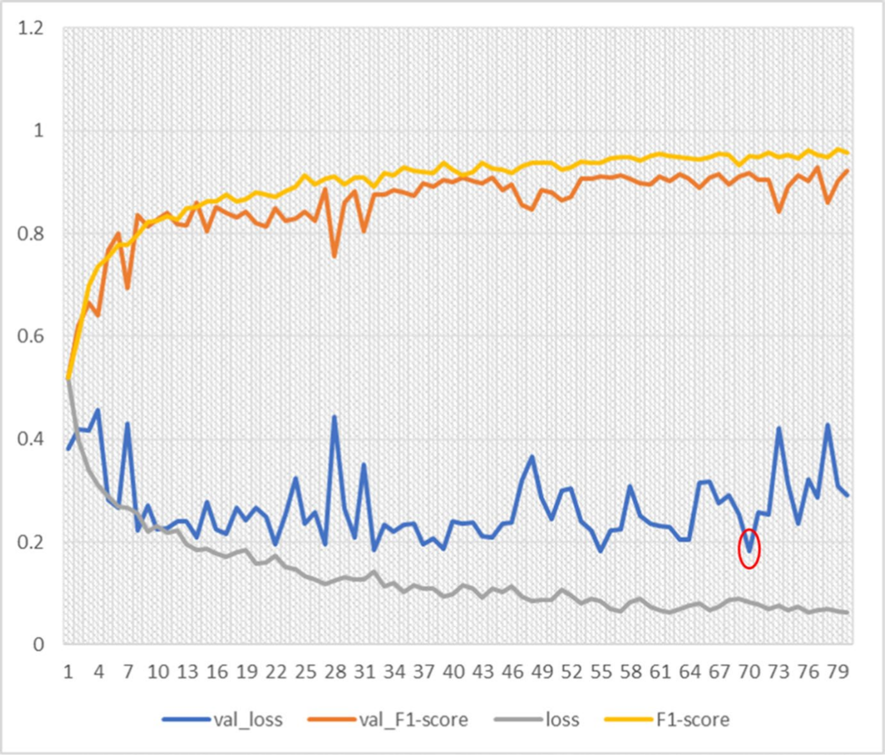


Fig. 6 The performance of CNN on webpage screenshots with 80 epochs

The comparative results are presented in Table 6.

From the table, it is found that the proposed method performs better than methods based on webpage screenshot in all the values. The precision and recall rate is more than 2.93% and 6.31% higher respectively and the F1-score rises by 4.68%, which further proves the effectiveness of SIA features which uses a much smaller image and saves computing resources and storage.

In short, from the above comparative experiments, the proposed CNN based SIA screenshot method takes fully consideration of visual counterfeiting characteristic of phishing websites, takes advantage of good performance of CNN on image classification and achieves the best result among these methods. In addition, besides saving labor and time (due to the advantage of deep learning algorithms mentioned above), the proposed

Table 6 Detailed results on the dataset of webpage screenshot at 70^{th} epoch

Categorization Methods	Precision	Recall	F1-score
Liu D & LEE J.-H.[34] (webpage screenshot)	0.953	0.888	0.919
The proposed method	0.981	0.944	0.962

method avoid the problems caused by spam techniques, various website languages and the easy change of texts phishing websites compared with methods using text information and also saves much storage space and computing resources compared to the method using webpage screenshots. It is very suitable for the use as browser plug-in, for browsers need to open websites anyway when users browse the web, which is all the method need once the model is trained and is applicable in realtity.

5 Conclusion and Future Work

This paper proposed a new phishing detection method that takes SIA screenshots and uses CNN as a classifier. It carries out comparative experiments on an unbalanced dataset with much fewer phishing sites between the proposed method and both classic machine learning and deep learning algorithms. The results have proved the better performance of the proposed CNN based SIA screenshot method. It is efficient and convenient to apply in larger and updated phishing detection tasks.

In future, we may make efforts in the following aspects: (1) developing a browser plug-in based on the proposed method and applying it in real Web environments; (2) fusing both visual and text features; and (3) applying and comparing more different algorithms.

Author Contributions Liu analyzed visual counterfeiting of phishing websites, and proposed the Security Indicator Area (SIA) as an input, which utilizes visual analysis and makes the input interpretable. Liu and Lee carried out several comparative experiments on a constructed unbalanced dataset. Liu and Lee reviewed the manuscript. Lee is the corresponding author of this paper.

Funding This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2021-0-00796, Research on Foundational Technologies for 6G Autonomous Security-by-Design to Guarantee Constant Quality of Security).

Declarations

Conflict of interest The authors have no conflicts of interest.

References

1. Tweneboah-Koduah, S., Skouby, K.E., Tadayoni, R.: Cyber security threats to IoT applications and service domains. Wireless Pers. Commun. **95**(1), 169–185 (2017)
2. Ponemon.: The Cost of Phishing & Value of Employee Training. https://info.wombatsecurity.com/hubfs/Ponemon_Institute_Cost_of_Phishing.pdf?t=1467214861789
3. NSFOCUS.: Phishing lecture hall Part2:Phishing risks (losses from attacks). http://blog.nsfocus.net/phishing-attack-risk/
4. Nirmal, K., Janet, B., Kumar, R.: Analyzing and eliminating phishing threats in IoT, network and other Web applications using iterative intersection. Peer-to-Peer Networking and Applications, pp. 1–13 (2020)
5. V, E.: Phishing Trends & Intelligence Report: The Growing Social Engineering Threat. https://info.phishlabs.com/2019-pti-report-evolving-threat

6. Microsoft.: Microsoft Security Intelligence Report Volume 24. https://info.microsoft.com/%20%20ww-landing-M365-SIR-v24-Report-eBook.HTML
7. Geng, G.G., Lee, X.D., Zhang, Y.M.: Combating phishing attacks via brand identity and authorization features. Secur. Commun. Netw. **8**(6), 888–898 (2015)
8. Chiew, K.L., Chang, E.H., Sze, S.N., Tiong, W.K.: Utilisation of website logo for phishing detection. Comput. Secur. **54**, 16–26 (2015)
9. Moghimi, M., Varjani, A.Y.: New rule-based phishing detection method. Expert Syst. Appl. **53**, 231–242 (2016)
10. Rao, R., Pais, A.: Detection of phishing websites using an efficient feature-based machine learning framework. Neural Comput. Appl. **01**(31), 3851–3873 (2018)
11. Jain, A., Gupta, B.B.: Towards detection of phishing websites on client-side using machine learning based approach. Telecommun. Syst. **12**(68), 687–700 (2017)
12. Sahingoz, O., Buber, E., Demir, O., Diri, B.: Machine learning based phishing detection from URLs. Expert Syst. Appl. **01**(117), 345–357 (2019)
13. Abbas, A., Singh, S., Kau, M.: Detection of Phishing Websites Using Machine Learning, pp. 1307–1314. Springer, New York (2020)
14. Gastellier-Prevost, S., Granadillo, G.G., Laurent, M.: Decisive Heuristics to Differentiate Legitimate from Phishing Sites. In: 2011 Conference on Network and Information Systems Security, pp. 1–9 (2011)
15. Geng, G., Yan, Z., Zeng, Y., Jin, X.: RRPhish: Anti-phishing via mining brand resources request. In: 2018 IEEE International Conference on Consumer Electronics (ICCE), pp. 1–2 (2018)
16. Zhang, X., Shen, C., Chen, Y., Wu, X., Liu, C.: An analysis of intelligent acousitic system. Front. Data Comput. **6**, 98–109 (2019)
17. Kreuk, F., Adi, Y., Cisse, M., Keshet, J.: Fooling end-to-end speaker verification with adversarial examples. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1962–1966 (2018)
18. Serdyuk, D., Audhkhasi, K., Brakel, P., Ramabhadran, B., Thomas, S., Bengio, Y.: Invariant Representations for Noisy Speech Recognition. In: 30th Conference on Neural Information Processing Systems (NIPS 2016) (2016)
19. Jiang, F., Fu, Y., Gupta, B.B., Liang, Y., Rho, S., Lou, F., et al.: Deep learning based multi-channel intelligent attack detection for data security. IEEE Trans. Sustain. Comput. **5**(2), 204–212 (2020)
20. Buczak, A., Guven, E.: A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surv. Tutor. **18**(2), 1153–1176 (2017)
21. Subasi, A., Molah, E., Almkallawi, F., Chaudhery, T.: Intelligent phishing website detection using random forest classifier. In: 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), pp. 1–5 (2017)
22. Parekh, S., Parikh, D., Kotak, S., Sankhe, P.: A New Method for Detection of Phishing Websites: URL Detection. In: 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), pp. 949–952 (2018)
23. Babagoli, M., Aghababa, M., Solouk, V.: Heuristic nonlinear regression strategy for detecting phishing websites. Soft. Comput. **02**(23), 4315–4327 (2018)
24. Rodríguez, J., García, V., Castillo, N.P.: Webpages Classification with Phishing Content Using Naive Bayes Algorithm, pp. 249–258. Springer, New York (2019)
25. Wei, B., Hamad, R., Yang, L., He, X., Wang, H., Gao, B., et al.: A deep-learning-driven lightweight phishing detection sensor. Sensors **09**(19), 4258 (2019)
26. Chen, W., Zhang, W., Su, Y.: Phishing Detection Research Based on LSTM Recurrent Neural Network. In: International Conference of Pioneering Computer Scientists, Engineers and Educators (ICPCSEE 2018) (2018)
27. Hiransha, M., Unnithan, N.A., Vinayakumar, R., Soman, K., Verma, A.: Deep learning based phishing e-mail detection. In: Proc. 1st AntiPhishing Shared Pilot 4th ACM Int. Workshop Secur. Privacy Anal.(IWSPA) (2018)
28. Cuzzocrea, A., Martinelli, F., Mercaldo, F.: A machine-learning framework for supporting intelligent web-phishing detection and analysis. In: IDEAS '19: Proceedings of the 23rd International Database Applications & Engineering Symposium (2019)
29. Alex, K., Ilya, S., Hg, E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of NIPS, IEEE, Neural Information Processing System Foundation. **01**(25), 1097–1105 (2012)

30. Liang, Y., Deng, J., Cui, B.: Bidirectional LSTM: An Innovative Approach for Phishing URL Identification, pp. 326–337 (2020)
31. Tajaddodianfar, F., Stokes, J., Gururajan, A.: Texception: A Character/Word-Level Deep Learning Model for Phishing URL Detection. In: ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2857–2861 (2020)
32. PhishLabs.: PhishLabs 2017 Phishing Trends & Intelligence Report. https://www.phishlabs.com/phishlabs-2017-phishing-trends-intelligence-report-hacking-the-human/
33. W3Techs.: Usage statistics of Default protocol https for websites. https://w3techs.com/technologies/details/ce-httpsdefault.2020
34. Liu, D., Lee, J.H.: CNN based malicious website detection by invalidating multiple web spams. IEEE Access **05**(8), 97258–97266 (2020)
35. Bisong E. In: TensorFlow 2.0 and Keras. Apress; 2019. p. 347–399
36. Manaswi, K.N.: Understanding and Working with Keras. Apress, pp. 31–43 (2018)
37. Liu, D., Lee, J.: CNN based malicious website detection by invalidating multiple web spams. IEEE Access **8**, 97258–97266 (2020)
38. Aljofey, J., Jiang, Q., Rasool, A., Chen, H., Liu, W., Qu, Q., Wang, Y.: An effective detection approach for phishing websites using URL and HTML features. Sci. Rep. **12**(1), 8842 (2022)
39. Lokesh, G.H., BoreGowda, G.: Phishing website detection based on effective machine learning approach. J. Cyber Secur. Technol. **5**, 1–14 (2021)
40. Alshehri, M., Abugabah, A., Algarni, M., Almotairi, S.: Character-level word encoding deep learning model for combating cyber threats in phishing URL detection. Comput. Electr. Eng. **100**, 107868 (2022)
41. Dilhara, S., Phishing, U.R.L.: Detection: a novel hybrid approach using long short-term memory and gated recurrent units. Int. J. Comput. Appl. **183**, 41–54 (2021)
42. Zhang, Q., Bu, Y., Chen, B., Zhang, S., Lu, X.: Research on phishing webpage detection technology based on cnn-bilstm algorithm. J. Phys. **1738**, 012131 (2021)
43. Al-Ahmadi, S., Lasloum, T.: PDMLP: phishing detection using multilayer perceptron. Int. J. Netw. Secur. Appl. **12**, 59–72 (2020)
44. Xu, P.: A Transformer-based Model to Detect Phishing URLs. J. Phys. Conf. Ser. (2021). arXiv preprint arXiv:2109.02138

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Dong-Jie Liu is currently with College of Cyber Security, Jinan University, Guangzhou, China. She received her Ph.D. degree from Computer Network Information Center, Chinese Academy of Sciences, University of Chinese Academy of Sciences, Beijing, China. Her research interests include Web security, machine learning, and blockchain. She has published papers in international authoritative journals and conferences. She also served as the program chair of ACM ICEA 2021 and technical committee member of ACM IECC 2020.

Jong-Hyouk Lee is now leading the Protocol Engineering Lab. at Sejong University. He carried his Ph.D. work in Computer Engineering from Sungkyunkwan University, Suwon, Republic of Korea. He received the IEEE Best Land Transportation Paper Award in 2015, Haedong Young Scholar Award in 2017, and IEEE Systems Journal Best Paper Award in 2018. He won official commendations from the Minister of Public Administration and Security, Republic of Korea in 2022 and from the Ministry of Science and ICT, Republic of Korea in 2023. He is an author of the Internet Standards: IETF RFC 8127, IETF RFC 8191, IETF RFC 8691, and IETF RFC 8818. His research interests include protocol engineering and security.