Deep Neural Network Based Model for Phishing-Sites Detection

San Kyaw Zaw, Khine Khine Oo
University of Computer Studies, Yangon
sankyawzaww@gmail.com, k2khine@gmail.com

Abstract

The evolution of web has positively transformed the paradigm of communication, trading, and collaboration for the benefit of humanity. However, these benefits of the Web are shadowed by cyber-criminals who use the Web as a medium to perform malicious activities motivated by illegitimate benefits. Phishing is a growing threat to Internet users and causes billions of dollars in damage every year. The replicas of the legitimate sites are created and users are directed to that web site by luring some offers to it. In this paper we introduce a model of our ongoing research Phishing Website Detection for Advanced Persistent Threats. In this model we used deep neural network technique on some features of phishing sites.

Keywords*: Phishing, URL Feature, HTML Feature, Model, Social Engineering, Security, Anti Phishing Technique, Data Mining, ANN,DNN and Phishing Attack.*

1. Introduction

The advanced persistent threat (APT) is an adversary that possesses sophisticated levels of expertise and significant resources which allow it to create opportunities to achieve its objectives by using multiple attack vectors. These objectives typically include establishing and extending footholds within the information technology infrastructure of the targeted organizations for purposes of infiltrating information, undermining or impeding critical aspects of a mission, program, or organization; or positioning itself to carry out these objectives in the future. The attackers employ techniques such as Social engineering and spear phishing to deploy malware into a network which steals all important operation data of the organization. In delivery stage of Advanced Persistent Threat (APT), attackers deliver their exploits to the targets by using two types of delivery mechanisms: spear phishing or watering hole [1]. Both of these techniques use malicious website called phishing websites. A malicious website may look identical to legitimate website in order to capture passwords, social security numbers, account numbers, and other confidential information [14].

Most of legit sites followed some standards that are set by the W3C, but a phisher may not follow these standards because of the intention of this site is to catch many fish in very tiny amount of bait and time. Based on the certain characteristics of the URL's and source code of the Phishing site, we can guess and determine to classify that the site is fake or not. Anti-phishing service providers like Google Toolbar employed various preventive strategies to detect and prevent the attacks from such phishing sites. These are the most common in the anti-phishing service providers [11].

The databases of blacklisted sites are creating and maintaining by these service providers. Some of the anti-phishing organizations maintain the blacklist of the reported phishing sites and their current status if they are still online or not, and Phishtank.com available as an example [2]. The techniques of maintaining online blacklist repositories fails when the phisher are creating sites at such a rate that there always will be some period in what the site is not reported as phish. The major drawback of this method is like the normal user will not always be taking caution about the phishing site, then victim may get tricked by overall look of site like legitimate site and it may happen like the site is not yet verified by the service providers and hence is not blocked.

Phishing website is a very complicated and complex issue to understand and to analyze, since it is a combination of technical and social dynamics for which there is no known single silver bullet to solve it entirely [12].

So, there is a need to create a resilient and effective intelligent model to detect phishing websites and to discover whether phishing activity is taking place or not. So far, various solutions have been proposed and developed to address these problems. Most of the previous approaches are giving rise to a large number of false positives, mainly due to limitation of such approaches, for example depending only on fixed black and white listing database, missing of human intelligence and experts, poor scalability and their timeliness [15]. Our ongoing research aims to create phishing website detection system that can prevent advanced persistent threat using the intelligence techniques. In this paper we introduce Deep Neural Network Based Model for Phishing-Sites Detection.

2. Previous Works

Phishing is a growing problem on the internet today for both consumers and businesses. One of the most common approaches for an attacker is to create a copycat website in order to capture personal information from consumers. A malicious website may look identical to legitimate website in order to capture passwords, social security numbers, account numbers, and other confidential information. The victim may not identify the malicious site until after the confidential information has been leaked. So, we review some of the existing phishing detection techniques as follow.

A. Jain and V. Richariya implemented a prototype web browser which can be used as an agent and processes each arriving email for phishing attacks. Using email data collected over a period time, they have presented an approach to detect phishing emails using link based features. The contribution of the work mainly consists of the usage of features visible links, invisible links and unmatched URLs [2].

M. A. Hossain et al. present an approach to overcome the ‘fuzziness’ in traditional website phishing risk assessment and propose an intelligent resilient model for detecting phishing websites in [11]. They used fuzzy logic operators to characterize the website phishing factors and indicators as fuzzy variables and produces six measures and criteria’s of website phishing attack dimensions with a layer structure. Website phishing detection rate is performed based on six criteria: URL & Domain Identity, Security & Encryption, Source Code & Java script, Page Style & Contents, Web Address Bar and Social Human Factor.

Rami M. Mohammad, Fadi Thabtah and Lee Mc Cluskey proposed a method “*Intelligent Rule based Phishing Websites Classification*”, a set of phishing websites was collected from Phishtank which is a free community site where users can submit, verify, track and share phishing data, Millersmiles which is considered a prime source of information about spoof emails and phishing scams[15]. The legitimate websites were collected from yahoo directory and starting point directory. They collected 2500 phishing URLs, and 450 legitimate ones. Then they extracted feature. To measure which features are significant in designing phishing websites, they calculated the frequencies for each feature in their datasets. They managed to collect and analyze 17 different features that distinguish phishing websites from legitimate ones. They performed experiments by using C4.5, RIPPER, PRISM and CBA algorithms.

Neda Abdelhamid deals with a challenging task making the number of existing algorithms for generating multi-label rules in associative classification (AC) from single label data sets by proposing an AC algorithm called Enhanced Multi-label Classifiers based Associative Classification (eMCAC) [12]. This algorithm discovers rules associated with a set of classes from single label data that other current AC algorithms are unable to induce. The proposed algorithm has been tested on a real world application data set related to website phishing.

3. Neural Network Vs Deep Neural Network

A standard neural network (NN) which is shown in Figure 1 consists of many simple, connected processors called neurons, each producing a sequence of real-valued activations. Input neurons get activated through sensors perceiving the environment; other neurons get activated through weighted connections from previously active neurons. Some neurons may influence the environment by triggering actions. Learning or credit assignment is about finding weights that make the NN exhibit desired behavior, such as driving a car.

Depending on the problem and how the neurons are connected, such behavior may require long causal chains of computational stages, where each stage transforms (often in a non-linear way) the aggregate activation of the network [8].

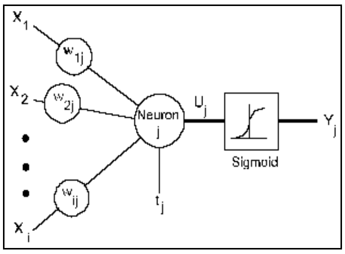

Figure 1. Neural Network

Deep Learning is about accurately assigning credit across many such stages. Since 2006, deep structured learning, or more commonly called deep learning or hierarchical learning, has emerged as a new area of machine learning research. During the past several years, the techniques developed from deep learning research have already been impacting a wide range of signal and information processing work within the traditional and the new, widened scopes including key aspects of machine learning and artificial intelligence [3].

Deep Learning means a class of machine learning techniques, where many layers of information processing stages in hierarchical supervised architectures are exploited for unsupervised feature learning and for pattern analysis/classification as shown in Figure 2. The essence of deep learning is to compute hierarchical features or representations of the observational data, where the higher-level features or factors are defined from lower-level ones. The families of deep learning methods have been growing increasingly richer, encompassing those of neural networks, hierarchical probabilistic models, and a variety of unsupervised and supervised feature learning algorithms [4][6].

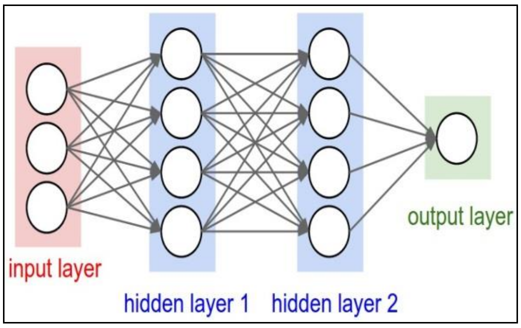

Figure 2. Deep Neural Network

4. Deep Neural Network Based Model for Phishing-Sites Detection

This section describes architecture and our approach towards the design of the system. The system architecture is shown in Figure 3.

Our model consists of two main components. The first component extracts different types of features from the web site. The second component is our deep neural network classifier which consists of an input layer, two hidden layers and an output layer. In the remainder of this section we describe each of these model components in detail.

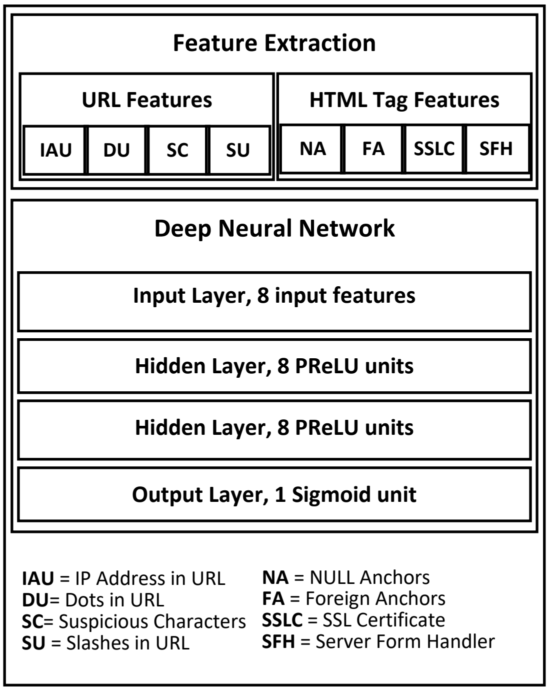

Figure 3 . Outline of Framework

4.1. Feature Extraction

The accuracy and efficiency of the model depends upon what features are to be used in the phishing detection. In this case, we have chosen four URL features and four HTML Tag features which can effectively determine the phishing attacks:

4.1.1. URL Features

(a) IP Address in URL: If an IP address is used as an alternative of the domain name in the URL, such as “http://125.98.3.123/fake.html”, users can be sure that someone is trying to steal their personal information. Sometimes, the IP address is even transformed into hexadecimal code as shown in the following link “http://0x58.0xCC. 0xCA.0x62/2/paypal.ca/index.html”. The domain needs to be registered in order to obtain a specific

URL address for the web site. Although legit sites have their domain registered and have the URL address, the phishing sites do last only for few days hence the phisher may not register the web site.

(b) Dots in URL: The dot in the URL represents the existence of the sub-domain in the URL. Let us assume we have the following link: http://www.hud.ac.uk/students/. A domain name might include the country-code top-level domains (ccTLD), which in our example is “uk”. The “ac” part is shorthand for “academic”, the combined “ac.uk” is called a second-level domain (SLD) and “hud” is the actual name of the domain. To examine this feature, we firstly have to omit the (www.) from the URL which is in fact a sub domain in itself. Then, we have to remove the (ccTLD) if it exists. Finally, we count the remaining dots. If the number of dots is greater than one, then the URL is classified as “Suspicious” since it has one sub domain. Some phisher may use the sub-domain to look the site address as the legit site hence causing the user to mislead to phish site. More the dots in URL, more the sub-domain existing in URL, intend to hide the web URL and look alike the legit site.

(c) Suspicious Characters: Using “@” symbol in the URL leads the browser to ignore everything preceding the “@” symbol and the real address often follows the “@” symbol. The dash symbol is rarely used in legitimate URLs. The Phisher will use some special characters other than alpha-numeric character to trick the user. To create the pattern of the legit URL which the user easily click on, special characters may be used ‘@’, ‘&’, ‘-’, and ‘_’ in the web URL.

(d) Slashes in URL: The slashes in the URL shows existence of sub-folders in it. The sub-folders are added to hide the information in the web-pages.

4.1.2 HTML Tags Features

(a) NULL Anchors: These are the Anchor tags in the web page which are not pointing to anything. When clicked on such links nothing happens or the links are redirected to the same page. After copying the source code of legit site the phisher may delete most of the links or replace with the link of the same page. More the NULL anchors are in page more the page is likely to be a phishing site.

(b) Foreign Anchors: These are the Anchor tags in web page which are linking to the domain which is not a domain or sub-domain of the current web site. Web sites can have some foreign links on it but too many foreign links will obviously increase the suspicion about that site. Phisher may copy the source code of legit web site to his own web page and then modify the web page in order to achieve the higher similarity with site he is trying to attack, the phisher cannot always modify each and every anchor tag which are pointing to legit sites and hence increasing the suspiciousness of that web page.

(c) SSL Certificate: It is nothing but Secure Socket Layer Certificate provided by some of the trusted authorities, the SSL Certificate covers the identity of the owner of the web page along with how it is encrypted and other information. Every legit page now has the SSL certificate 2.0 or 3.0 versions. The SSL Certificate has validity of very short period and needs to be updated over period of time. Most of the browsers allow the web page access when SSL Certificate is present. As the SSL Certificate is only provided to legit and verified owners of web pages, phisher has very less chances of obtaining SSL Certificate.

(d) Server Form Handler (SFH): SFHs that contain an empty string or “about: blank” are considered doubtful because an action should be taken upon the submitted information. In addition, if the domain name in SFHs is different from the domain name of the webpage, this reveals that the webpage is suspicious because the submitted information is rarely handled by external domains.

4.2. Neural Network

For classification, we use a deep feedforward neural network consisting of four layers, where the first three 8 node layers consist of a dropout [11], followed by a dense layer with either, a parametric rectified linear unit (PReLU) activation function [9] in the first two layers, or the sigmoid function, in the last hidden layer (the fourth layer being the prediction).

Dropout has been demonstrated to be a very simple and efficient approach for preventing overfitting in deep neural network. Unlike standard weight regularizers, such as based on the $\ell 1$ or $\ell 2$ norms, that push the weights toward some expected prior distribution [13], dropout seeks weights at each node that are complementary to weights in other nodes. This can be viewed as implicit bagging of several neural network models [10].

Rectified linear units (ReLU) have been shown to significantly speedup network training over traditional sigmoidal activation functions, such as tanh [3], by avoiding significant decay in gradient descent convergence rate after an initial set of iterations. This slowdown is due to saturating non-linearities in sigmoidal functions at their edges [3] [7] [9].

4.2.1. Formal Description

Let l = {0, 1, 2, 3} be a layer in the network, y(l−1) the incoming values into the layer (for l =1 those are the feature values), y(l) the output values of the layer, W (l) the weights of the layer that linearly transforms n input values into m output values, bl the bias, and F(l) the associated activation vector function. The equation for *l* = {1, 2, 3} of the network is

$$d^{(l)} = y^{(l-1)} \cdot r(l),\quad z^{(l)} = W^{(l)} d^{(l)} + b^{(l)},\quad y^{(l)} = F(z^{(l)}), \qquad (1)$$

where · is a pointwise (elementwise) vector product, and r_i are independent samples from a Bernoulli distribution with parameter *h*. The r values are resampled for each batch update during training, and *h* corresponds to the fraction of nodes that are kept during each batch update [11]. Layer *l* = 0 is the input layer, and *l* = 4 is the output layer. For layers *l* = {1,2}, the activation function is the PReLU function,

$$F(z_1^{(l)}) = (y_1^{(l)}, \ldots, y_i^{(l)}, \ldots, y_m^{(l)}) \qquad (2)$$

where for some additional parameter $a_i^{(l)}$ that is also fit during training,

$$y_i^{(l)} = \begin{cases} a_i^{(l)} z_i^{(l)} & \text{if } z_i^{(l)} < 0, \\ z_i^{(l)} & \text{else.} \end{cases} \qquad (3)$$

For the final layer *l* = 3, the activation function is the sigmoid function,

$$y^* = \frac{1}{1 + e^{-z^{(3)}}}, \qquad u(4)$$

which produces the output of our model. The loss for each n sized batch sample is evaluated as the sum of the cross-entropy between the neural net's prediction and the label,

$$L(y^*, \hat{y}) = -\sum_{j=1}^{n} [\hat{y}_j \log y_j^* + (1 - \hat{y}_j) \log(1 - y_j^*)] \qquad (5)$$

where y^* is the output of our model for all n batch samples, y_j^* is the output for sample j, and $\hat{y}_j \in \{0,1)$ is the label of the sample j, with 0 representing legal site and 1 phishing site. The neural network is training using backpropagation and the Adam gradient-based optimizer [5], which we observed to converge significantly faster than the standard stochastic gradient descent.

5. Estimated Results

Based on the training set taken from UCI Machine Learning Repository, some predictions are evaluated and proposed model is verified. Before training, we perform normalization process to input the feature into our system. Following example shown in Table 1 will illustrate how the system will predict the results by the proposed model.

Table 1. Example for Prediction

WB	IAU	SC	SU	DU	SSL	NA	FA	SFH	Lab
A	0	1	0	0	0	0	1	0	0
B	1	1	1	1	1	1	1	1	1
C	1	0	1	0	1	1	0	1	1
D	1	1	1	0	0	0	0	0	0
E	1	1	1	0	1	0	0	0	0
F	0	1	0	0	1	1	1	0	1
G	1	1	0	1	1	1	1	1	1
H	1	1	0	0	0	0	0	1	0
I	1	0	1	0	1	1	0	0	1
J	1	1	0	0	0	0	0	0	0

As shown in Table1, we used 8 of the features for testing purpose as our full system is under construction. We haven't tested the whole dataset for the estimated result. After constructing the full system, we intended to use various techniques such as precision, recall and f-score to check the accuracy of our system. We expected to get the acceptable accuracy of over 80%.

6. Conclusions

In this paper we introduced a deep learning based phishing site detection approach. Our approach requires modest computation to perform feature extraction and that it can achieve good accuracy. Neural networks also have several properties that make them good candidates for phishing site detection. It can allow incremental learning, thus, not only can it be training in batches, but it can retrained efficiently (even on an hourly or daily basis), as new training data is collected. Moreover, through pre-training of individual layers, it allows us to combine labeled and unlabeled data. Additionally, prediction can be done very quickly using low amounts of memory because of the classifiers are very compact.

References

[1] “Advanced Persistent Threats: A Symantec Perspective: Preparing the Right Defense for the New Threat Landscape”, Symantec Corporation, 2011.

[2] A. Jain, V. Richariya, “Implementing a Web Browser with Phishing Detection Techniques”, 2011.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks”, In Advances in neural information processing systems, pages 1097–1105, 2012.

[4] A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models”, In Proc. ICML, volume 30, 2013.

[5] D. Kingma and J. Ba. Adam, “A method for stochastic optimization”, arXiv preprint arXiv: 1412. 6980, 2014.

[6] G. E. Hinton, S. Osindero, and Y.W. The, “A fast learning algorithm for deep belief nets”, Neural computation, 18(7):1527–1554, 2006.

[7] J. Friedman, T. Hastie, and R. Tibshirani, “The elements of statistical learning”, volume 1.Springer series in statistics Springer, Berlin, 2001.

[8] J¨. Schmidhuber, "Deep Learning in Neural Networks: An Overview", Technical Report IDSIA-03-14 / arXiv:1404.7828 v4 [cs.NE] (88 pages, 888 references), October 2014.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”, arXiv preprint arXiv:1502.01852, 2015.

[10] L. Deng and D. Yu, "Deep Learning: Methods and Applications", Foundations and Trends in Signal Processing Vol. 7, Nos. 3–4 (2013) 197–387, 2014.

[11] M. Aburrous, M. A. Hossain, F. Thabatah, K. Dahal, “Intelligent Phishing Website Detection System using Fuzzy Techniques”, 2010.

[12] N. Abdelhamid, “Multi-label rules for phishing classification”, 2014.

[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting”, The Journal of Machine Learning Research, 15(1):1929–1958, 2014.

[14] P. Chen, L. Desmet, Christophe Huygens, “A study on Advanced Persistent Threats”, 2014.

[15] R. M. Mohammad, F. Thabtah, L. M. Cluskey, “Intelligent Rule based Phishing Websites Classification”,2012.