**Phishing categorization**

**Final report**

**Deep Learning**

**Abstract:**

Phishing is a type of social engineering attack often used to steal data, including login credentials and credit card numbers. In the web, it occurs when a malicious web site impersonates a legitimate one in order to acquire sensitive information. In this paper, we developed deep-learning based methods for detection of phishing sites. Our research suggests a multi-layered anti-phishing solution that aims to use a set of features, image based detection for important websites and also the ability to prevent users from leaking pre-defined confidential information. A dataset of on approximate 22000 active phishing sites and 22000 non-phishing sites will be used in this comparative study. We conclude with thoughts on future for such techniques and what could have been done with better resources.

1. **Introduction:**

Phishing is a form of social engineering. Its goal is to psychologically manipulate people into performing actions or divulging confidential information. The way the attack is performed may vary depending on prior knowledge about the victim and the attacker setup. For instance, one form of phishing attack is when a web site impersonates a legitimate one in order to steal sensitive information such as password, user-name, credit card, etc. Another form of phishing attack can be through emails, where the attacker tricks their target into sending confidential information or clicking a link to a malicious website. Nowadays, phishing is the number one delivery vehicle for ransomware and other malwares [1].

Anti-Phishing Working Group (APWG) has shown a constant increase in phishing activities as well as the related cost. According to Forbes, phishing attacks cost American businesses half a billion of dollars a year [2]. In the first half of 2017, APWG reported that the number of unique phishing website detected were 291,096[3]. Phishing attacks affects millions of Internet users around the globe, and become a significant threat to user and businesses alike.

Over the past few years, much attention has been paid to the issue of security and privacy. Many are dealing with phishing directly. Unfortunately, most of those solutions are designed for the enterprise market and when applied on individuals they suffer an unacceptable rate of false positives.

The contribution of this study is to suggest a solution aims to provide a better security for individuals from phishing attacks. We will focus on preventing phishing attacks in the web. We suggest a multi-layered solution: (1) Building a classifier to predict if a certain website is a part of a phishing attack. (2) Using deep learning to identify phishing website by using visual attributes such as favicons.

The study is using a phishing data-set [4] that is consist of 25,311 online, valid phishing websites. Same size data-set of the “normal” websites was selected randomly from the list of the most popular websites provided by Amazon [5]. In addition, a self-scrapped dataset of icons which consists of almost 6000 logos of 30 leading websites.

The rest of the paper is organized as follows: In section 2 we discuss related work. section 3 – project description. In section 4 we describe our previous attempts. Section 5 described our experiments and results.

1. **Related Works:**

The proposed solution is using many features that has already been studied before. These features can be categorized as follows: (1) Machine learning approach on the URL. (2) Machine learning approach on the website content or visual

Machine learning approach on URL:

A study from “Southern Methodist University” [6] described a detailed comparison of machine learning techniques for phishing detection. The study evaluates logistic regression, Regression trees, Random forests, Neural Networks, Bayesian Additive Regression Trees (BART) and more. Although, the research fails to declare a winner. It suggests ways to evaluate and compare between the different approaches correctly.

Machine learning approach on content:

A common factor to almost every successful phishing attack in the web, is that it looks like a different website that we are regular to use. Such as a fake “Facebook” login page. In this approach, the phishing detection is done by examining site contents. A study from “Drexel University” [7] tried to detect phishing by looking at visual similarities. The showed that this technique result in a great detection rate, but suffers from slowness.

1. **Project Description**

This research offers two techniques that can be used together to help prevent phishing attacks appearing in the web. We used two techniques, one based on the complete URL of the website and second trying to identify the use of the original website icon. Together, it can provide additional layers of defense for individuals browsing the web.

1. **URL classifier:**

We have built a URL classifier that aims to identify between phishing websites to “good” websites. The research is using a phishing data-set [4] that is consist of 25,311 online, valid phishing websites. Same size data-set of the “normal” websites was selected randomly from the list of the most popular websites provided by Amazon [5].

The classifier is using three features:

1. URL length – From observing the data we noticed that sometimes phishing URLs length is much longer than an average “known good” URL.
2. Number of dots – Phishing websites has to manipulate the user into thinking that website is legitimate. Using dots in strategic places in the URL can trick users into thinking it is a completely different website. (Example: face.book.com)
3. Number of digits – Same as dots, digits can also mislead users.

Using this data and these features we create a Multilayer Perception (MLP) classifier. The settings of the MLP classifier:

1. Relatively large “alpha” for the gradient decent – Our compute time doesn’t allow many iterations of gradient decent and as a result, the coefficients we were looking for were close to the random numbers we gave them initially. We were able to reduce the loss function much faster.
2. We chose a threshold of 0.4 to get the best results from our classifier. See figure 1.

Figure (1) – The MLP classifier threshold

1. We used 3 hidden layers.
2. **Icon detector:**

We also created a visual based phishing recognition. Identifying that a webpage is similar to a different webpage has been proven by related work to be a difficult task also that takes a lot of time to determine (if a website is good or not).

We decided to take a different approach and identifying if the logo of the websites appears in strategic places in the website while the URL is different than it should be. For example, a website that uses the Facebooks favicon, while its URL is different than facebook.

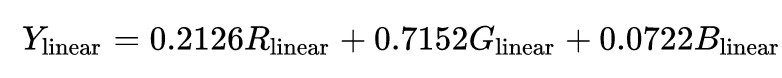
In order to do so, we used Convolutional neural network (CNN).

The data we used was a self-scrapped data from a well-known icon website [8]

We have collected 6000 icons of 30 different websites such as Facebook, Google, Twitter, Outlook, etc.

We have converted the PNG formatted icons to 16x16 pixels grayscale image.

Since the image was originally in RGBA format we used the following formula to convert the pixels to grayscale [9]:



See figure 2.

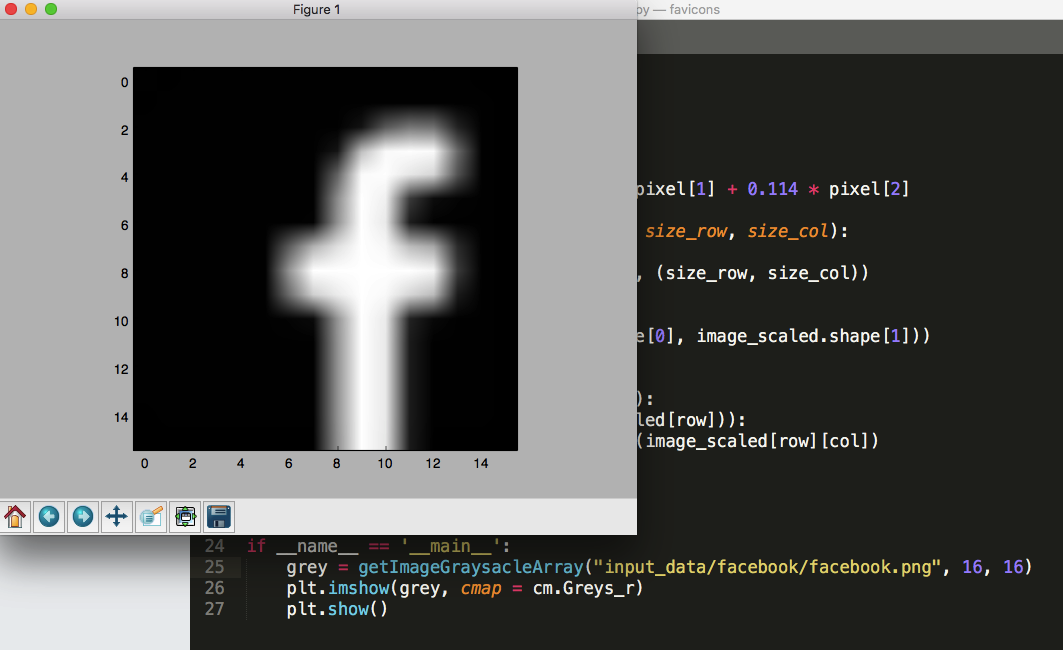


Figure (2) – 16x16 grayscale image

The CNN we have built is structured (Figure 3) as follows:

1. Convolutional layer followed by max-polling – The convolutional layer computed 32 features for each 5x5 patch. Afterwards applying ReLU layer and Max-Pooling (2x2) reducing the image to 8x8 size.
2. Second convolutional layer followed by max-polling – Same as above with 64 features for each 5x5 patch. This process will reduce the image size to 4x4.
3. Fully-Connected layer – Now that the image is of size 4x4 we add a fully connected layer with 1024 neurons to allow processing the entire image.
4. To reduce overfitting, we have added a dropout layer.

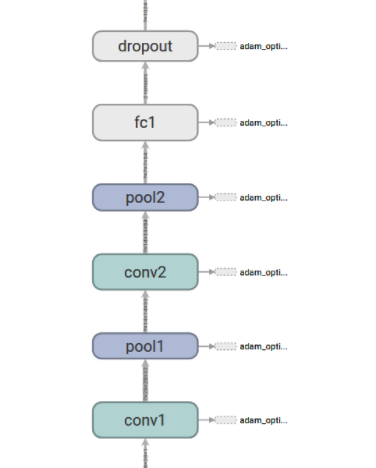


Figure (3) – The CNN structure we used

Instead of using Gradient decent to minimize the loss function we used ADAM optimizer.

1. **previous attempts**

It worth mentioning that using Logistic Regression was also part of our study.

First, we have used our URLs data-set to study URLs using Logistic Regression (A single layer of MLP).

We used the same features and the same data-set. As expected we have gotten much better results using MLP.

1. **Experiments / Simulation Results**

The experiment we did to test our two models was as follows:

1. 80% of the data was used to train the model
2. 20% training data was used to validate the group
3. 20% out of the training data was used as a sanity check to our models.
4. MLP results:

When training on the data, we’ve noticed that we were able to reduce the loss function significantly see figure 4.



Figure (4) – MLP loss function and bias

When running on the “sanity check” data we have received 73% of accuracy, also known as “train error”. When running on the validation group we received 68% of accuracy.

1. CNN results:

We had a similar experiment to the CNN model.

When running on the “sanity check” data we received 99% of accuracy.

When running on the validation group we received 96% of accuracy.

1. **Conclusions**

This research has shown the effectiveness of machine learning and deep-learning techniques to identify phishing attacks. Although the MLP classifier was failed to be accurate even on the training data, we think that using a rich set of features and advance machine learning techniques can improve this model significantly and should be done in future studies. We would like to extend our visual model (CNN) to identify other parts of the website (such as login forms) to enhance our protection.

1. **Resources**

[1] **Must-Know Phishing Statistics 2017**

Jonathan Crowe - <https://blog.barkly.com/phishing-statistics-2017>

[2] **Phishing Scams Cost American Businesses Half A Billion Dollars A Year**

Lee Mathews in *Forbes* - https://www.forbes.com/sites/leemathews/2017/05/05/phishing-scams-cost-american-businesses-half-a-billion-dollars-a-year/#680a1e683fa1

[3] **Phishing Activities and Trends Report**

Greg Aaron – *APWG* - <http://docs.apwg.org/reports/apwg_trends_report_h1_2017.pdf>

[4] **Phishing URL Data-set** - <https://www.phishtank.com>

[5] **Known Good Website Data-set -** <http://s3.amazonaws.com/alexa-static/top-1m.csv.zip>

[6] **A Comparison of Machine Learning Techniques for Phishing Detection**

Saeed Abu-Nimeh , Dario Nappa , Xinlei Wang , and Suku Nair - *SMU HACNet Lab Southern Methodist University* -<https://docs.apwg.org/ecrimeresearch/2007/proceedings/p60_abu-nimeh.pdf>

[7] **PhishZoo: Detecting Phishing Websites By Looking at Them** Sadia Afroz

Rachel Greenstadt - Department of Computer Science Drexel University - <http://www1.icsi.berkeley.edu/~sadia/papers/phishzoo-icsc_final.pdf>

[8] **Icons Data-set -** <http://iconfinder.com>

[9] **Grayscale -** https://en.wikipedia.org/wiki/Grayscale#Converting\_color\_to\_grayscale