2022 International Conference on Computer Communication and Informatics (ICCCI -2022), Jan. 25-27, 2022, Coimbatore, INDIA

MALICIOUS WEBSITE NAVIGATION PREVENTION USING CNNs AND URL VECTORS: A STUDY

SIDDHANT TIWARI
Computer Science
Undergraduate
SRM Institute of Science and Technology, Kattankulathur
Chennai, India

HAIDER RIZVI
Computer Science
Undergraduate
SRM Institute of Science and Technology, Kattankulathur
Chennai, India

DR. K. KALAISELVI
Assistant Professor
SRM Institute of Science and Technology, Kattankulathur
Chennai, India

Abstract: In this paper, we have focused on the problem of malicious URLs. URL attacks have been on the rise in 2020, with most of the work being online based due to the pandemic there arises a greater scope of Phishing URLs etc.
There have been existing systems but they are mostly paid, whereas with this project we aim to deploy a freemium add-on in a web browser, hosted on cloud with a real time dynamic classified URLs database so as to make the process more accessible and at the same time, less CPU and RAM consuming.
The main highlights of our thesis have been that the accuracy measures of the two mains algorithms have been really close but there are discrepancies in the confusion matrix itself. Although these differences arise because of the time bindings and we would face such problems while deploying this project as an add-on service on a browser, a slow-fast multilayered system seems a better prospective plan to pursue in the future.

I. INTRODUCTION

The world has seen immense development in the field of Internet technology which in turn with its many benefits has caused some problems as well and the biggest being - diverse threats to network security. Attackers have been found spreading malicious Uniform Resource Locators (URLs) to initiate attacks like phishing, spam and financial fraud for gathering sensitive information or capital. The prevention of malicious URL attacks requires more research to be performed for accurate and efficient detection. WWW has advanced widely and is being constantly employed by companies for the purpose of malware distribution.

A Kaspersky Catalogue from 2020 tells us that 85% of threats on the web were due to disinfected URLs.[1] Around ⅓ of all existing URLs are found to be malicious in another research.
Malicious URLs can harm internet users in various ways ranging from phishing attacks for cyber frauds, spyware installations, ransomwares, RATs[2] etc.

Previously techniques like blacklisting regular expression and signature matching were applied but they were ineffective and inaccurate. Blacklisting is the maintenance of an updated database of URLs already confirmed to be malicious. But due to URL generators the risks can't be entirely negated and hence blacklists can be avoided.
This brought the involvement of machine learning techniques and some other methods which would basically work on heuristic feature extraction principles.

With this project, our plan is to compare a model based on Convolutional Neural Network (CNN) to be used for malicious URL detection along with models that use feature extraction principles.

A CNN is a deep learning algorithm which uses

978-1-6654-8035-2/22/$31.00 ©2022 IEEE

artificial neural networks with different layers in V The other significant technique utilized are feature extractions based on URL vectors which are described in IV.

II. RELATED WORKS

The classification of URLs has utilized a variety of algorithms since the dawn of url phishing.

This section gives a brief description of related work in the detection of malicious webpages. In the past few years, research efforts have been made on the detection of malicious URLs using data mining approaches.

BINSPECT [**3**] was a lightweight approach presented by Eshete et al, that combines static analysis and emulation to detect malicious web pages by applying supervised learning techniques which achieved a 97% accuracy and had a low rate of false signals.

Machine learning classifiers have statistical methods that were explored by Ma et al. [**4**] to detect malicious URLs based on lexical and host-based features of URLs. Classifiers obtained 95-99% accuracy. This experiment proved to be highly accurate but it also consumed a lot of time which cannot be converted into real time systems as they need to be fast.

Also Curtsinger [5] used Bayesian classification(ZOZZLE) whereas techniques like multi-label classifier and rule mining were put to good use.

WebMon is constructed using the random forest learning algorithm [6] because the combination of these two resulted in the best accuracy.

Safe Browsing by google is another blacklisting method to detect malicious URLs.

Cao has used an opposite of the Google Safe Search that is an Automated Individual White-List (AIWL). This white list uses a Naive Bayes classifier and keeps record of the website visited by the users and notifies them when there is a possibility of an attack.

But again the possibility of new URLs via dynamic URL generators curb the purpose of these systems.

III. WHAT ARE URLS

The Full-form of URL is Uniform Resource Allocator. As the name suggests an URL is a location or address of any resource or information over the internet. These resources can be anything - a video, a HTML document, an audio file etc. URLs reference these sources. All the websites that are visited by people using their browsers are also URLs.

The URL consists of two types of parts - the first one defining the type of protocol, the second part holds the domain name, then comes the path, parameters and the source. After the protocols there's a double slash //. The path and parameters use the separators like a single slash, question marks etc. while the domain name is separated from these using a dot(.).:-

A. *Components of URL*

In all the URL can be said to be composed of 6 parts namely -

1) Scheme - e.g. http / https
2) Domain Name - e.g.www.google.com, www.amazon.in etc.
3) Port - Usually 80 for the internet
4) Path to the file - The location of the file on the server on which the website is hosted on.
5) Parameters - This is an optional part of the URL and is generally used when the website needs a runtime input from the user.
6) Anchor - The address to a particular part of the document that the user needs to visit.

B. *How to use URLs*

The user can type the URL they might want to navigate to on their browser's address bar to go to the website and access the resource it stores within itself. But, this is just the start of the internet experience.

The HTML language the only language that the browser understands as website uses URL in its syntax as follows:

Media can be displayed on the web pages using HTML <img>,<video> and <audio> tags.

Other documents on the webpage can be linked using the <a> tag.

IV. URL FEATURES

Features must be extracted from URLs to obtain vectors.

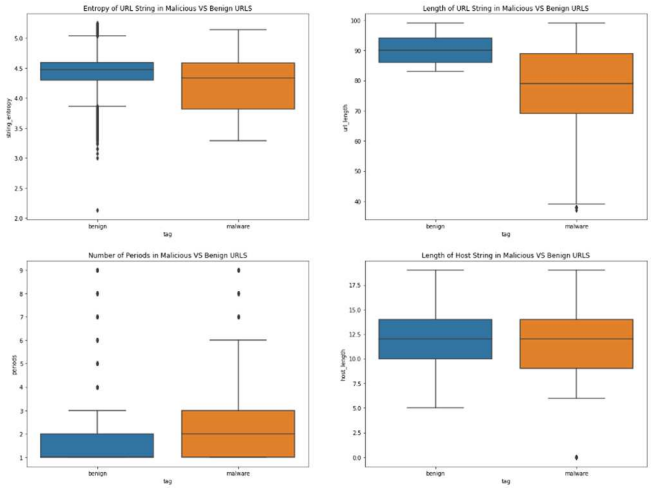
URLs constitute basically two types of features: -

- Lexical
- Host Based
- Content Based Features

Lexical features - These are the features derived from the name of the URL. It covers how the URL manifests or displays itself in public, and many harmful URLs may try to impersonate the benign ones. These may be combined with host based features to improve model performance and score.
Statistical information in the URL counts as a part of lexical features. Features like token counts in domain name, path name, longest/average domains or paths, special characters used are compiled into a dictionary where each word is a feature. Presence and absence are indicated via 0 or 1.
Entropy is another feature which measures the randomness of a domain name. Some Malicious URL obfuscation methods utilize Domain Generation Algorithms, which manipulate the urls consistently. Thus by setting limits on URL entropy helps govern URL legitimacy.
Sensitive words are also extracted as a part of lexical features (login, sign in, secure etc.)
In recent times, researchers have found advanced lexical features to be useful when calculated with significant heuristics in mind.
Another interesting feature is Kolmogorov Complexity[7].
These advanced features concentrate on - URL related, Domain related(length, IP etc.), Directory Related, File names and Argument features.

Host based Features -
Host Based Features as the name suggests are characteristics derived from host-names.
Hackers started utilizing the exploits that come through with short urls, and the Time To Live was very less for URLs to be detected and classified.
These provide other information like webpage, ports, connection speed etc.

Content Based Features:
These are the features associated with the download of the webpage/
They are obtained from HTML and JavaScript information but these features require a lot of extraction.

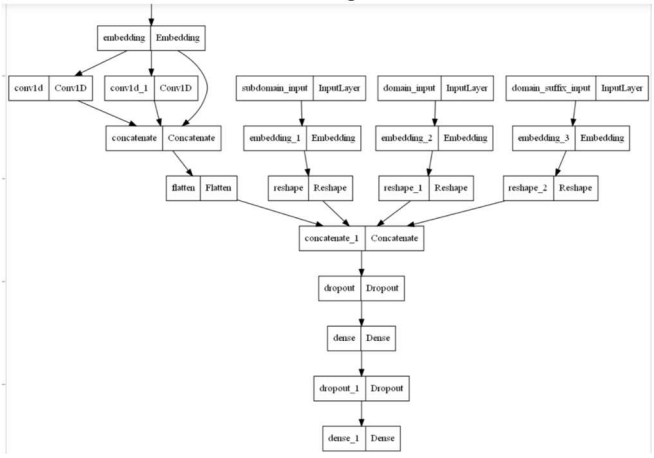


Lexical features comparison plots[8]

V. OVERVIEWS OF CNNS

A CNN is a deep learning algorithm which uses artificial neural networks with different layers. Filters are applied to input layers and an activation is obtained.
CNNs take inputs, assign importances via biases and weights and then develop differences between the inputs in our case which are URLs.
Convolutional Neural Networks (CNNs) are similar to ANNs[9] and contain neurons which optimize according to self learning. Each neuron will still receive an input and perform an operation.

The CNN model is used first and the predictions and validations are made against the 20% of data.



VI. DEEP LEARNING MODELS

A. CNN - LSTM

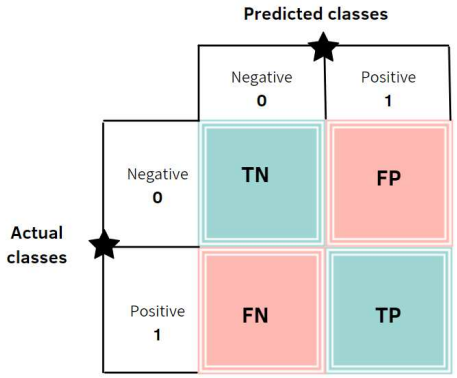
The CNN LSTM (Long Short-Term Memory) Network for is an LSTM architecture that is created

for problems that require sequence prediction with spatial inputs for example videos, images etc. These were basically developed for visual time series problems and for generating textual inferences from a sequence of images i.e. videos.
The algorithm works with a CNN that has been trained before on a challenging image classification. The LSTM architecture has also been used on speech recognition and natural language processing problems.

VII. EVALUATION METRICS

We utilize general ML metrics such as Confusion matrices, precision, recall and accuracy for our thesis.

Confusion matrices are as follows



True Positives (TP): Correct malicious URLs prediction.

True Negatives (TN): Correct benign URLs prediction.

False Positives (FP): Incorrect malicious URLs prediction.

False Negatives (FN): Incorrect benign URLs prediction.

Precision and recall are two more parameters utilized.

$$\text{Precision} = \frac{\text{True Positive}}{\text{Actual Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive + False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{Predicted Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

Precision is the proportion of positive identifications which were truly predicted.

Recall is the proportion of actual positives that were identified correctly.

$$\text{Accuracy} = \frac{\text{True Positives + True Negatives}}{\text{All Samples}}$$

Accuracy is the ratio of correct predictions to total samples.

VIII. RESULTS

In this paper we have studied various methods for the evaluation of websites and their classification as malicious or benign.
We have not utilized a new feature but we have evaluated the existing methods and used a variety of datasets [10] for their result calculation. The empirical results show the efficiency of the various models. After using the functions we've made like parsed_URL and extract_URL along with the python module tldextract, we obtain and convert the dataset in this form.

	url	label	subdomain	domain	domain_suffix
0	mister-ed.com/welcome/file/update/rbc/login.php	bad	0	0	0
1	ip-23-229-147-12.ip.secureserver.net/public/fi...	bad	1	1	1
2	facebok-info.com/unitedkingdom/log.php	bad	0	2	0
3	independent.co.uk/news/obituaries/john-gross-g...	good	0	3	2
4	facebook.com/geoffrey.gray	good	0	4	0

The comparison that has been done here is between a convoluted neural network to detect malicious URLs and URL vectors to detect malicious internet addresses present in the database. Two different databases were utilized to get to the results.
The convoluted neural networks had the following output -

```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     69148
           1       1.00      0.99      0.99     20888

    accuracy                           1.00     90036
   macro avg       1.00      0.99      1.00     90036
weighted avg       1.00      1.00      1.00     90036
```

As it is evident from the classification report that the CNN model provides us with great accuracy. Also looking at the confusion matrix the false

prediction for both the positive and negative values is low as compared to the correct predictions.

```
Confusion Matrix:
[[69115    33]
 [  217 20671]]
```

Now, coming to the URL vectors, the output received are as follows -

```
#Decision Tree
dt_model = DecisionTreeClassifier()
dt_model.fit(x_train,y_train)

dt_predictions = dt_model.predict(x_t
accuracy_score(y_test,dt_predictions)
```

```
0.9954970107005496
```

The accuracy achieved here using the URL vectors is quite similar to the one received with the CNN model but the confusion matrix can be seen as being not as good as the one seen in CNN.

```
print(confusion_matrix
```

```
[[241238    714]
 [   705  72467]]
```

So, for conclusion we can observe that we get a better confusion matrix with CNN model.
But the time factor and the space factor pull down the CNN model.
URL vectors trained model is a better fit to be deployed as an online service.
The best way to deploy an online detection service for malicious URLs would be a combination of Blacklist and URL Safe browsing techniques from which if a URL is missed, it then is passed to the Feature vector or the CNN model.

IX.REFERENCES

[1]Kaspersky. Malware Variety Grows by 13.7% in 2019 Due to Web Skimmers. Available online: **https://www.kaspersky.com/about/press-releases/2019_malware-variety-grows-by-137-in-2019-due-to-web-skimmers** (accessed on 20 January 2020)

[2] Remote Access Trojans are programs that provide the capability to allow covert surveillance or the ability to gain unauthorized access to a victim PC. https://blog.malwarebytes.com/threats/remote-access-trojan-rat/

[3]Eshete, B.; Villafiorita, A.; Weldemariam, K. BINSPECT: Holistic Analysis and Detection of Malicious Web Pages. In *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering*; Springer: Berlin, Germany, 2013; Volume 106 LNICS, pp. 149–166.

[4] Ma, J.; Saul, L.K.; Savage, S.; Voelker, G.M. Beyond blacklists: Learning to detect malicious web sites from suspicious URLs. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2009; pp. 1245–1253.

[5]Curtsinger, C.; Livshits, B.; Zorn, B.; Seifert, C. ZOZZLE: Fast and precise in-browser JavaScript Malware detection. In Proceedings of the 20th USENIX Security Symposium, San Francisco, CA, USA, 8–12 August 2011; pp. 33–48.

[6] Kim, S.; Kim, J.; Nam, S.; Kim, D. WebMon: ML- and YARA-based malicious webpage detection. Comput. Netw. 2018, 137, 119–131.

[7] Hsing-Kuo Pao, Yan-Lin Chou, and Yuh-Jye Lee. 2012. Malicious URL detection based on Kolmogorov complexity estimation. In Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01. IEEE Computer Society

[8]Ruth Eneyi Ikwu https://towardsdatascience.com/extracting-feature-vectors-from-url-strings-for-malicious-url-detection-cbafc24737a

[9] Artificial neural networks (ANNs), usually simply called neural networks (NNs), are computing systems inspired by the biological neural networks that constitute animal brains https://en.wikipedia.org/wiki/Artificial_neural_network

[10] Dataset sources -1 https://www.unb.ca/cic/datasets/url-2016.html

-2 https://www.kaggle.com/sid321axn/malicious-urls-dataset