**Implication of Machine Learning Approach in Detecting Phishing Urls**

Saugatdeep Ranabhat

Moscow Institute of Physics and Technology (MIPT)

[Moscow,](mailto:rami.mohammad@hud.ac.uk) Russia

**Abstract**

Phishing is an online scam where an attacker sends out phoney messages that appears to be from a reliable source, as a result of such scam attacker can access a vast amount of data. This type of attack can be discovered in a variety of ways, one of which is machine learning. The URLs that the user receives will be entered to the machine learning model, which will then process the input and present the results, indicating whether they are phishing or not. To achieve this goal, the research's mission is to analyse and compare benign and phishing websites using data from a chosen dataset. With the help of dataset and the url details we will extract meaningful features out of urls which will be a starting point of our research. Later with those features we will be using six most popular machine learning (ML) models, including AdaBoost, Linear Discriminant Analysis, K Nearest Neighbor, DecisionTree, RandomForest, and ExtraTree to check their efficiency in detecting malicious urls. At last we will conclude our project with a comparative performance evaluation with ML model against an Artificial Neural Network algorithm to find out the best performing model.

### Keywords: Phishing, Url Feature Extraction, Machine Learning, Safe web browsing.

### Introduction

A growing number of everyday activities are supported by the web, which is a popular platform. Since many people have experienced the leakage of personal and corporate data, data security has recently been a hot topic. Attackers seek to pose as a recognised person or organisation at the same time. They use a variety of platforms to entice customers, such as compromising a legitimate website or adding alluring promotions or pop-ups to unofficial community forums or messaging. Phishing is the term for such fraudulent activities. Simply put,  phishing is a type of threat to faculty data in which phishers purposefully target a person or organisation. The development and advancement of organizations crossing across numerous applications including internet banking, online business, and interpersonal interaction because of the appearance of new correspondence innovations. With the help of internet, malware or various cyber-attacks are launched. Conveying malicious content on the web has turned into a standard method for attackers because of expanded web access by the greater part of the total populace. Social-engineering, denial/distributed denial of service (DOS/DDOS), phishing, man-in-the middle, SQL injection, theft of devices and numerous others are assortment of methods used to execute cyber attack.

The impediments of conventional security the executives advancements are turning out to be increasingly more serious given this dramatic development of new security dangers, fast changes of new IT innovations, and huge deficiency of safety experts. Drive-by download, Phishing, Spam, Social Engineering are most well-known kinds of attacks utilizing malicious URLs. To keep away from information robbery or spillage URL distinguishing proof is by all accounts the best arrangement. The rising number of security events and huge economic losses warn us that distinguishing malicious url has become a great challenge for both individuals and enterprises who need to protect their data security and maintain integrity. Cyber security researchers always tend to look for a roubust approach to recognize and take action against such threats. Machine learning (ML), a method of data analysis that automates analytical model building is one of such effective approach that can be utilized in detecting phishing urls. ML has two goals, one is to classify data based on models which have been developed, the other purpose is to make predictions for future outcomes based on these models. Machine learning procedures go through a few stages and distinguish the pernicious URLs in an exact way. This strategy additionally gives the insights concerning the false positive rate.

Therefore, this paper focuses on exploring the behaviours of malicious URLs and extracting their features in three different dimensions that can distinguish malicious from benign URLs based on employed Neural Network and ML algorithms. Here performance and accuracy of Artificial Neural Network (Multilayer Perceptron, MLP) and ML classifier are compared. The remainder of this paper is organised as follows: Section 2 summarises the related work about the detection methods, Section 3 describes the proposed approach and introduces the classification models. Section 4 presents the system implementation and the experimental results. Section 5 will conclude the paper by outlining possible directions for future study.

### Related Work

Techniques utilized in phishing detection field have been changing over the long run as specialists continue to deal with to fabricate a powerful and more successful discovery component and to tackle advance attacks by so called phishers. Besides, there were different studies done in phishing field. The fundamental techniques for recognizing malevolent URLs are blacklist, heuristic and machine learning method. These are explained as follows.

**2.1. Blacklist Approach**

One of the most widely recognized and classical method for identifying malicious URLs is Blacklist approach. As per Jian Zhang et al. [[1]](#JianZhang) this method has a database that comprises of a bunch of URLs that are malevolent in past. This technique is exceptionally quick and simple to carry out. At the point when a user visits a new URL, then a quest for database is acted in blacklist. In the event that the new URL falls in blacklist category, an admonition will be created to show that the URL is malicious otherwise it's a benign one. This strategy can have exceptionally low false positive rates. Bad actors making difficulties on Blacklist Approaches. S. Ganera et al. [[2]](#SGarera) recognizes mostly four kinds of entanglements in this methodology. The first is that utilizing an IP it makes disarray on host. Second justification for inconvenience is mistaking the host for different domains. Third one is making disarray on host by utilizing enormous host names. Another challenge in blacklist approach is found by the new expansion in the critical degree of URL in an impressive sum i.e., the URL is made considerably shorter and direct into a required page.

S. Sinha et al. [[4]](#four) proposes a reputation-based blacklist approach. With this, compromised hosts as well as malignant contents in URLs, host and network can be determined. Then utilizing this data access of web, emails and different actions in pernicious organizations or host can be impeded. The fundamental disadvantage of this approach is keeping up with thorough rundown of malicious URLs, which is extremely challenging.

**2.2.** **Heuristic or Rule-based Approach**

The blacklist method lacks ability to recognize recently produced malicious URLs so an enhancement technique is utilized by C. Seifert et al. [[6]](#six) that is the Heuristic approach. This method is essentially utilized for distinguishing the malignant sites by isolating phishing site features. This technique makes a blacklist of signatures at whatever point another URL is shown up. Then it is investigated with the accessible rundown of marks. In the event that there is a match of that sort, the URL is alluded as malicious. In this methodology, a signature is allotted out to every one of the perceived normal attacks in view of its behavior.

The Nguyen et al. [[12]](#twelve) explains heuristic approach in his paper. As per him, the heuristic-based detection method examines and extract characteristics of phishing sites and based on that information detects phishing sites. In order to detect whether a requested site is a phishing site, first extracts feature from the URL in the user-requested page. It aids in minimising the harm brought on by phishing attempts. There are two modules for the heuristic technique for detecting malicious URLs, according to Nida Khan1 et al. [[13]](#thirteen). The first module in this method is the module that matches URLs and DNS records. This module has a white-list like blacklist method. White-list usage can control an increase in runtime and a drop in the false negative rate. It has two parameters: an IP address and a domain name tie-in. When a user enters a website, the system looks for a match with the domain name of the currently-visited website and adds it to its whitelist. If a website is on a whitelist, users can access it; otherwise, the system compares the IP address of the matching domain to determine whether a DNS poisoning attempt has occurred. The white list's initial value is zero due to the absence of a domain in the list. The white-list grows whenever a person accesses a new website or types in a new URL. The most crucial aspect is that when a person accesses a website or URL for the first time, the domain of that URL will not be on the white-list. The identifying module is where the second module begins. By removing hyperlinks from the web page and using phishing detection algorithms like ID3 or C4.5, this module may identify phishing web pages or URLs. The module then determines whether the URL of the web page is malicious or benign. Warning will be triggered if found malicious. The drawback of the heuristic approach is that it consumes a lot of system resources and launches an attack right away once whenever a website is visited. As a result, the attack that is generated could go unnoticed.

**2.3**. **Machine Learning Approach**

In the first two sections, shortcomings of the blacklist method and the heuristic approach were discussed. To overcome these issues, majority of researchers began focusing on machine learning (ML) and using these approaches to distinguish between malicious and benign URLs. According to [[14]](#fourteen), artificial intelligence may have a strong demand for machine learning. These methods offer a system that can learn on its own and get better with time without a set of programme. The primary objective of machine learning is to give a system the ability to learn autonomously without the involvement of humans. According to [[15]](#fifteen), supervised learning, unsupervised learning, and semi-supervised learning are the key machine learning methodologies. A mentor provides direction during the supervised learning process. This method take some data set and act as a mentor. It directs the model or trains the machine. After proper guidance it starts to take decisions on the newly fed data. They are able to learn independently (without the help of mentor) through unsupervised learning. I.e., based on observations and a certain data structure. This method operates differently from supervised learning. By extracting patterns and relationships from a given collection of data, this approach creates clusters. Between supervised and unsupervised learning is semi-supervised learning [[14]](#fourteen). For training, it employs both labelled and unlabeled data. This machine learning technique offers superior performance and accuracy.

1. **Approach**

Up until this section, we discussed about how authors have contributed for a robust approach in detecting phishing urls. As shown in Fig.1. represents the complete flow of our proposed system from data collection, feature extraction and training/testing approach for classification purposes and the output consisting of accuracy and classification result. First and foremost, collection of reliable and informative dataset is a very important aspect in dealing with learning based problems for classification or regression. The data consisting of both malicious and non-malicious URLs with labels need to be collected for training and testing purpose from a reliable source in order to get better accuracy and classification result. We have chosen a dataset [[16]](#nineteen) that has sufficient amount of data consisting of both legitimate and phishing urls. It contains total 450k urls out of which 345k are legitimate and 104k are malicious.

Now that the dataset contains only the list of urls and it’s label, next step is the extraction and selection of features which are sufficient for the description of URLs and which can be mathematically interpreted for training using Machine Learning models. In Machine Learning, a feature is an individual measurable property or characteristic or an attribute of a phenomenon being observed. Choosing informative, differentiating and independent features is a vital step for efficient algorithms in pattern recognition, classification and regression purposes. Simply, using an URL will not directly allow good classification method. So, it is important to select suitable features based on some rules or hypothesis to obtain a good feature from the set of URL. Thus, the quality of the extracted features from the URLs is prominent for the quality of the resulting malicious URL classification model.

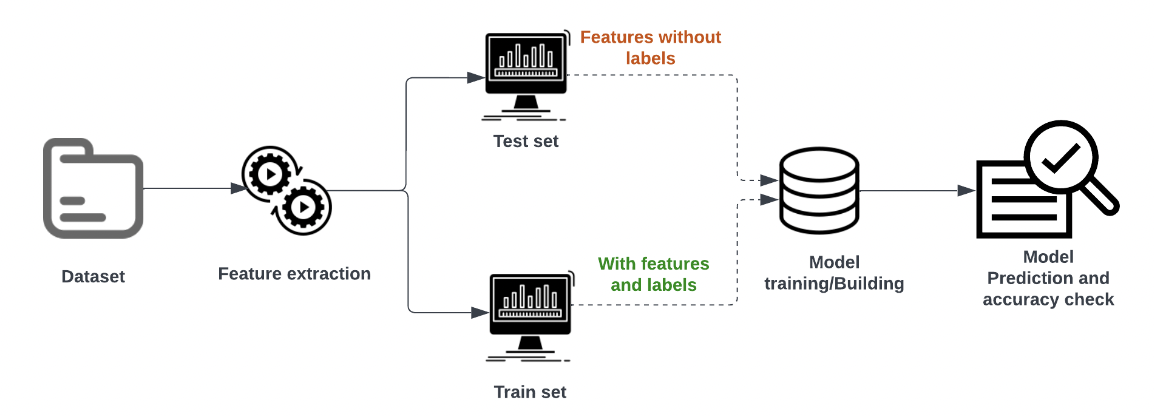


Fig.1. Phishing url detection framework

* 1. **Feature Extraction**

Feature extraction is at the core of the processing; it identifies the essential characteristics of malicious web traffic to distinguish malicious URLs from normal the benign. We have extracted total 19 features for each url present in our dataset, these features are then fed into an algorithm that builds the detection model. Below mentioned category of features are extracted from the URL data and same is shown in Fig.2 :

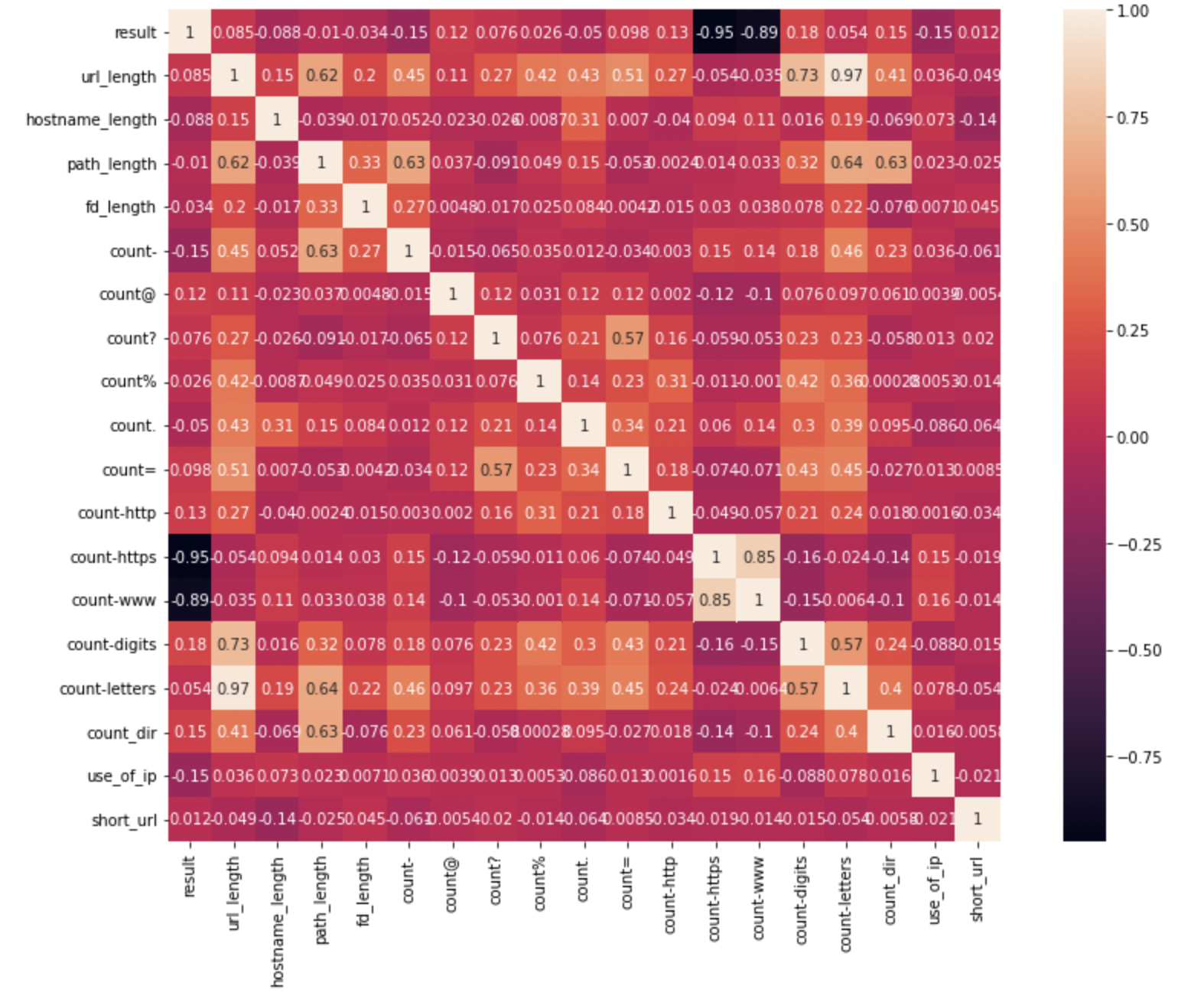


Fig.2 Heatmap of extracted features.

1. URL feature extraction : Length based Features (4 features extracted)

* Length Of Url
* Length of Hostname
* Length Of Path
* Length Of First Directory

Considering the following URL we can illustrate the extraction of length based features of a url: **http://www.urlfeatures.com/dir1/predict.php**

Given URL is 43 characters long, thus the length of url is 43. The length to the path “dir1/predict.php” is 16 and so on.

1. Count based Features (12 features extracted)

As the name suggest, count based features are extracted with reference to the number of relavent prefix or resource that are involved in a url.

* Count Of '-'
* Count Of '@'
* Count Of '?'
* Count Of '%'
* Count Of '.'
* Count Of '='
* Count Of 'http'
* Count Of ‘https’
* Count Of 'www'
* Count Of Digits
* Count Of Letters
* Count Of Number Of Directories

1. Binary Features (2 features extracted)

* Use of IP: If IP address present in URL then the feature is set to 1 else set to 0. Most of the benign sites do not use IP address as an URL to download a webpage. Use of IP address in URL indicates that attacker is trying to steal sensitive information.
* Use of Shortening URL: TinyURL service allows phisher to hide long phishing URL by making it short and the primary goal is to redirect user to phishing websites. If the URL is crafted using shortening services (like bit.ly) then feature is set to 1 else 0.
  1. **Classification algorithm**

A classification issue, more specifically a "binary" classification issue, is the issue we are attempting to resolve. Classification issues fall under the category of supervised machine learning. Following feature extraction, we train several ML models using our dataset and select the one that provides the best accuracy. The fundamental concept behind detection is to increase the detection accuracy by feeding accurate, clean data to the machine learning classification model. Machine learning classification models have good performance and could be applied; therefore, determining how best to process the web data is the most significant procedure. The accuracy and performance of different machine learning algorithms can vary depending on the type of data provided, so we must train different algorithms and evaluate their performance before selecting the one model that performs best. We will be using six ML and one ANN algorithms with each algorithm having it’s own significance in terms of performance and classification property. AdaBoost, Linear Discriminant Analysis, K-Nearest Neighbor, Decision Tree, Random Forest and Extra Tree classifier are used from ML category whereas Multilayer Perceptron; an artificial neural network algorithm is used at the end to compare the accuracy in classifying malicious and bening urls with respect to the before mentioned ML models.

* 1. **Evaluation procedure**

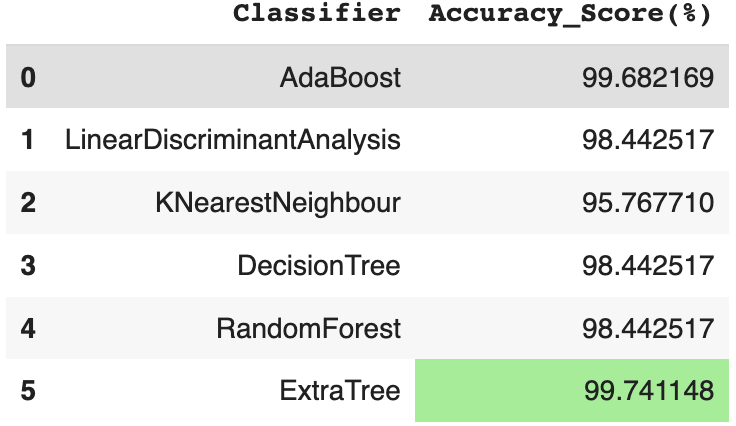
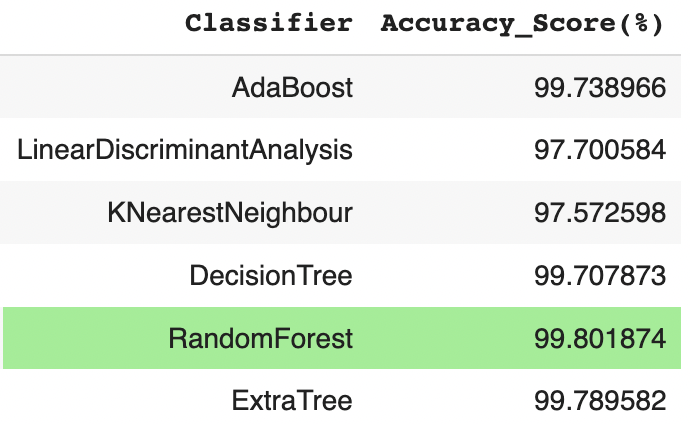
Prior to deploying the detection model for actual use, algorithms should be tested to see how well they work. Key statistical indicators are already present in machine learning procedures, so we can use those as evaluation metrics to simultaneously choose the best algorithm. We can use the training model in a production environment if the precision and recall ratios are satisfactory; if not, we return to the previous step and retest. The issue can frequently be brought on by the process of gathering or preparing data. In that case, the preprocessing stage needs to be reviewed. In conclusion, developing an accurate evaluation procedure is challenging because the outcome can be affected by each step. Additionally, each step might have room for improvement. The best performance be determined only by performing numerous evaluations and tests.

1. **Experiment and results**

The expirement is divided into two parts, first we will utilize our six machine learning models: AdaBoost, Linear Discriminant Analysis, K-Nearest Neighbor, Decision Tree, Random Forest and Extra Tree classifier. An additional experiment is performed with the use of Multilayer perceptron as our neural network model to compare the performance metrics against traditional machine learning approach. The goal of the experiment is to see which classifier performs the best in detecting malicious urls.

In our dataset of 450K records, there were only three labels i.e url, label of the url and result (mapping benign as 0 and malicious as 1). With feature engineering total 19 crucial url features were generated which were saved for the purpose of model training and testing.

Dataset was split into training and testing sets in the form of 80% and 20% ratio respectively. First we applied all above mentioned algorithms to train and test our models and the results are shown in Fig 3. As per the results, ExtraTreeClassifier has the highest accuracy among the rest. At this point we decided to rerun train/test with same classifiers but this time with a balanced dataset as during our EDA we noticed that our dataset is imbalanced. Just to find out the difference in performance between balanced and imbalanced dataset in classification we applied Synthetic Minority Oversampling Technique (SMOTE); an oversampling technique where the synthetic samples are generated for the minority class. After the oversampling process, we applied our ML algorithms and outcome of it is shown in Fig 4.

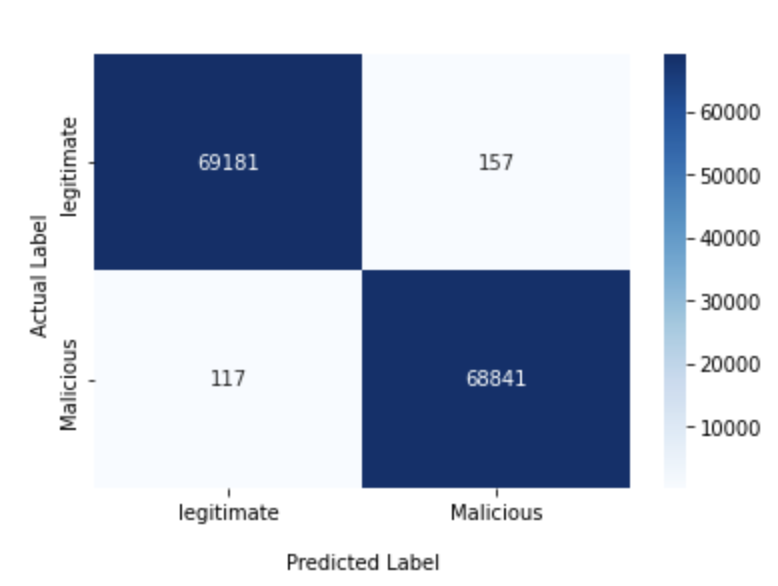
Comparing the results of Fig 3. and Fig 4., we can observe that with the balanced dataset we can achieve better accuracy from our ML classifiers.

**Fig.4. Accuracy result with balanced dataset**

**Fig.3. Accuracy result with imbalanced dataset**

Here, accuracy is defined as the percentage of correct decisions among all testing samples and accuracy is computed with reference to formula (1).

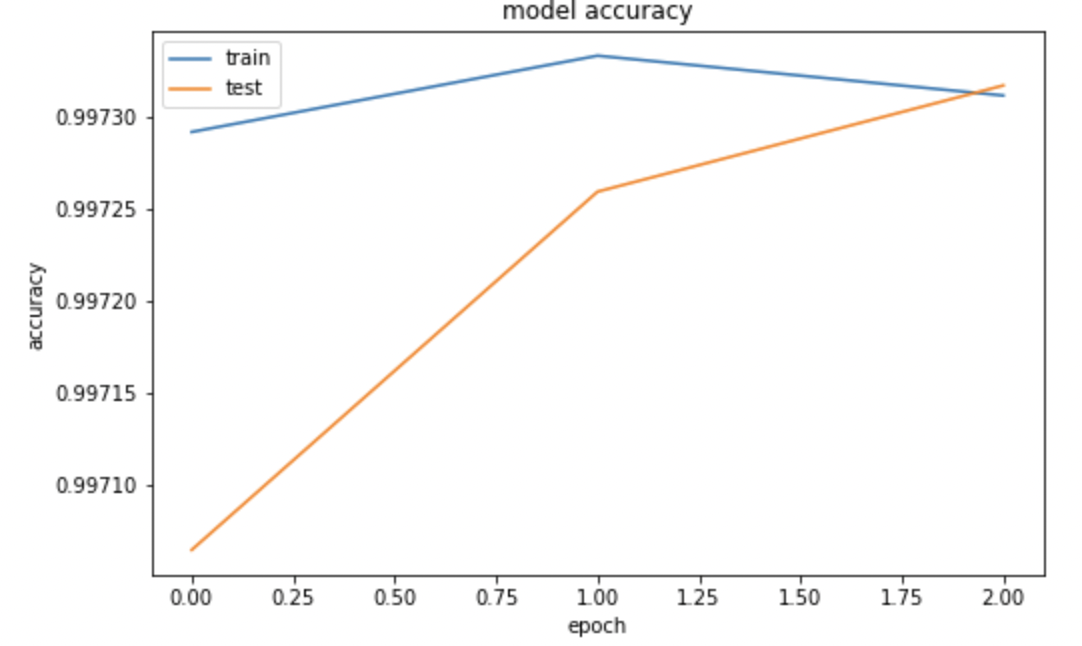
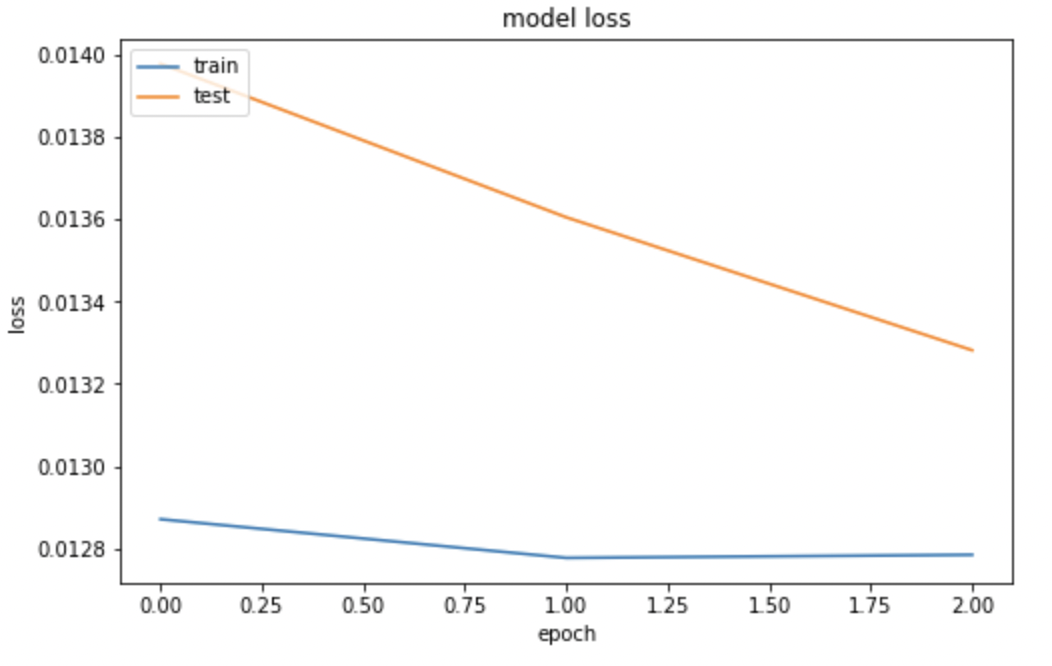
whereby TP, TN, FP, and FN denote true positive, true negative, false positive and false negative respectively.



**Fig.5.Confusion Matrix of Random Forest**

Other performance metrics evaluated with the proposed system are precision, recall and F1-score using formula (2), (3) and (4) respectively The classification accuracy, ACC is used as the evaluation metric, and is computed as follows:

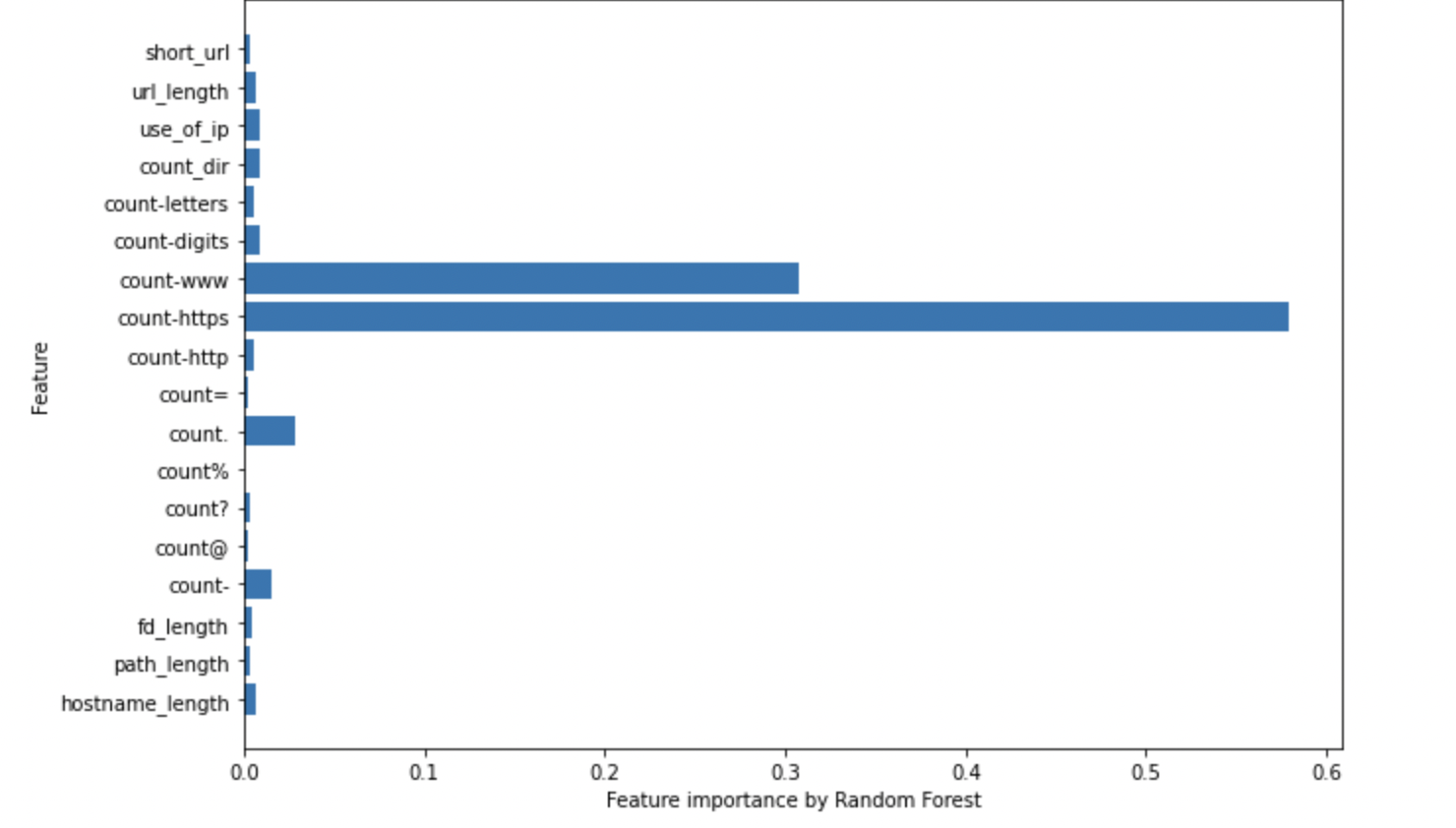
After ML classifiers, we then applied multilayer perceptron (MLP) to train/test our dataset to see whether it can produce more better results than our best performing ML model. A MLP has input and output layers, and one or more hidden layers with many neurons stacked together. And while in the perceptron the neuron must have an activation function that imposes a threshold, like ReLU or sigmoid, neurons in a MLP can use any arbitrary activation function. With our MLP model we got an accuracy score of 99.56 % which still is on the lower side than our best classifying ML model i.e. random forest. Performance details of MLP is illustrated in Fig 4 and Fig 5.



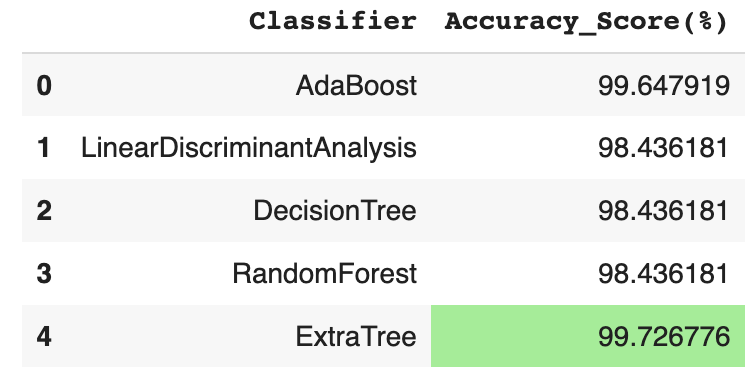
**Fig 5. MLP model accuracy**

**Fig 6. MLP model loss**

Based on testing results, we can conclude that Random forest classifier has the highest accuracy in detecting phishing urls with only modest false positives and it is obvious that url features as shown in Fig 7. plays a key role in alalyzing a url and it’s serverity.



**Fig 7. URL feature importace**



1. **Conclusion**

Compared with traditional detection methods, this system can accurately detect malicious URLs that have fuzzy characteristics. The detection model does not rely on any one characteristic or on a blacklist but is a comprehensive approach that utilises all the features for the machine learning. This work explored the applications of machine learning and deep learning models in detecting and classifying malicious URLs. By collecting performance accuracy, confusion matrices, and both training and predicting times for all classifiers, this study concludes that Random Forest gave the highest accuracy (99.8%) with suitably balanced precision and recall and is considered the best model to deploy by organizations seeking to build an URL filter application, or those wishing to incorporate machine learning techniques to improve existing ones. Additionally, this study presented some unique way of extracting features from Urls for more accurate and detailed analysis.

In future work, the detection can be made more specific by also analysing the behaviours of malicious URLs. The precision may be improved to 99.99% by perfecting the malicious key words and extracting more features.

**Refrences:**

[1] Jian Zhang, Phillip A. Porras, and Johannes Ullrich. Highly predictive blacklisting. In Proceedings of the 17th USENIX Security Symposium, July 28-August 1, 2008, San Jose, CA, USA, pages 107122, 2008.

[2] S. Garera, N. Provos, M. Chew, and A. D. Rubin, A framework for detection and measurement of phishing attacks, in Proceedings of the 2007 ACM workshop on Recurring malcode. ACM, 2007, pp. 18.

[4] S. Sinha, M. Bailey, and F. Jahanian, Shades of grey: On the effectiveness of reputation based blacklists, in Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on. IEEE, 2008, pp. 5764.

[6] C. Seifert, I. Welch, and P. Komisarczuk, Identification of malicious web pages with static heuristics, in Telecommunication Networks and Applications Conference, 2008. ATNAC 2008. Australasian. IEEE, 2008, pp. 9196.

[7] P. Gutmann. The Commercial Malware Industry., 2007.

[8] KALPA, Introduction to Malware, 2011.

[9] G. Jacob, H. Debar, and E. Filiol, Behavioral detection of malware: from a survey towards an established taxonomy, Journal in Computer Virology, pp. 251266, 2008.

[12] Nguyen, Luong Anh Tuan,”A novel approach for phishing detection using URL-based heuristic.” Computing, Management and Telecommunications (ComManTel), 2014International Conference on. IEEE, 2014.

[13] Nida Khan, Manliv Kaur, Riddhi Panchal, PrashantKumar Rai, Nilesh Rathod Heuristic Based Approach for Fraud Detection using Machine Learning InternationalJournal of Innovative Research in Computer andCommunication Engineering, Vol. 6, Issue 2, February 2018.

[14] Marco Varone, Daniel Mayer, Andrea Melegari https://www.expertsystem.com/machinelearning-definition/

[15] Atul Published on Jan 09, 2019 https://www.edureka.co/blog/what-ismachine-learning/

[16] https://www.kaggle.com/datasets/siddharthkumar25/malicious-and-benign-urls