Journal of Electrical Engineering, Electronics, Control and Computer Science – JEEECCS, Volume 9, Issue 33, pages 39-54, 2023

Phishing Website Detection Using Machine Learning

Model development and Django

Seun Mayowa Sunday
ADEY Innovations Limited
St. Modwen Park, Stonehouse, Gloucester, United Kingdom
rightmayowa@gmail.com

***Abstract* – The increasing number of phishing attacks is one of the major concerns of security researchers today. Traditional solutions for spotting phishing websites rely on signature-based methods, which cannot detect newly generated phishing websites. Thus, researchers are developing machine learning-based systems capable of detecting and classifying phishing websites with high accuracy, given a vast and diverse set of data. After several steps that require adequate preparation of the dataset for the model development, the prepared dataset is used to train the logistic regression (LR), k-nearest neighbor (KNN), and artificial neural network (ANN) model. This research is concluded by integrating the best-performing model in terms of the documented measuring metrics into the Django application. Research has proved that the integration of the machine-learning model into the web application is lacking. Researchers only stop at the model performance without proper integration into the end-user consumption. Apart from the comparison of the proposed model with previous researchers' work, this research will also contribute by detailing the steps required to integrate the proposed model for end-user consumption using the Django Framework.**

Keywords- Security, Phishing, Traditional-based methods, ML, LR, ANN, KNN, Django

I. INTRODUCTION

According to the X-Force Threat Intelligence Index, phishing was the most common technique employed by cybercriminals to gain access to a business or organization [1]. In the majority of situations, they do so in order to start a much wider attack, such as ransomware, network infiltration, etc. X-Force neutralized 41% of phishing attacks in 2021, according to the findings of the Index [1].

Using sophisticated methods, tactics, and technologies, such as phishing via content injection, social engineering, online social networks, and mobile applications, phishing assaults are meant to collect sensitive information. Despite the fact that there are numerous additional definitions of phishing, the Anti-Phishing Working Group [2], provides the following definition: Phishing is defined as "an illicit practice involving the use of social manipulation and technical deceit to obtain individuals' personal information and login credentials." This APWG is a global alliance composed of 2200 institutions from all over the world. The APWG publishes annual trend reports about phishing activity. According to a report published in 2020, 34% of attacks targeted consumers of software-as-a-service (SaaS) and webmail providers. In addition, after the COVID-19 breakout in March 2020, there was a considerable increase in the number of global phishing attacks [2]. It is also interesting to note that 75% of phishing websites use SSL (Secure Sockets Layer), thus we can deduce that the SSL protocol does not always bring us to a trustworthy website.

The vast majority of the time, however, this process begins with an email that scares consumers into taking immediate action. In the majority of situations, phishing attacks commence with an email. Beyond email, phishing attempts can target social networks, blogs, forums, VoIP, mobile applications, and messaging systems [8]. In recent years, phishing attacks that target a range of systems, including blockchain platforms, have emerged. Following the all-time market high valuations of cryptocurrencies like Bitcoin and Ethereum, scammers have switched their focus to these digital assets. These attacks may result in not just monetary loss, but also the loss of sensitive user data and intellectual property (IP). It also has the ability to erode trust and national security [6]. Consequently, phishing protection is more important than ever before.

Utilizing a trustworthy browser should be regarded as the initial line of defense against phishing attacks [9]. The browser defense techniques utilize blacklists provided by platforms such as Phish Tank, Safe Browsing, and SmartScreen. It is also possible to identify phishing attempts using specialized security software such as an Intrusion Detection System (IDS) or an Intrusion Prevention System (IPS).

The problem with the denunciation platforms is that a zero-day phishing attack, which is an attack related to a newly constructed phishing site, cannot be identified because it will not be on the blacklist for a long time [5]. Due to the short lifespan of phishing websites and the rapid creation of new ones, the work required to

manage blacklists is excessive [3]. In addition, managing blacklists requires an excessive amount of effort. Changing a single character in the URL of a website can make it invisible to blacklists [14]. Because certain types of phishing assaults, such as spear phishing, target just specific corporations and individuals, these websites may not be up on blacklists [7].

A number of obstacles were encountered with the blacklist-based detection technique, which led to the creation of heuristics-based alternatives [9]. To categorize websites as either fraudulent or authentic, a prediction model is constructed utilizing a variety of variables extracted from the website's URL and page content. A substantial number of researchers have thus far employed a variety of machine learning algorithms for the detection of phishing [11].

Multiple institutions' researchers presented a variety of categorization strategies for distinct phishing detection techniques. [4] classified countermeasures as machine learning (ML), deep learning (DL), scenario-based techniques (ST), and hybrid strategies (HT). Deep learning is a subsection of ML that automates the discovery of features and the development of end-to-end prediction systems. It accomplishes this by analyzing vast quantities of data [10]. When analyzing ST, a lot of different scenarios are considered, and attacks are discovered with the use of these scenarios. The goal of HT approaches is to provide improved results in terms of evaluation criteria for accuracy, precision, and recall by combining multiple methods.

Recent advancements in supervised learning have provided cutting-edge answers to numerous research difficulties, including face recognition and image categorization. Additionally, they were effectively applied to a number of cybersecurity difficulties, including the identification of malware, the detection of intrusions, the detection of spam emails, and the detection of phishing sites [15]. Although there have been implementations of machine learning algorithms for the detection of phishing, there are, to the best of the authors' knowledge, no academic paper that documents the step-by-step integration into the web application. As a result, this research will contribute by providing a methodical approach to the detection of phishing using machine learning and step by step integration into the Django web framework.

1.1 Scope of this Study

Although model building is achievable when appropriate datasets exist, many cyber security industries do not make their datasets available. In light of this, the Canadian Institute of Cyber Security dataset [26], which is freely accessible for educational purposes, is utilized in this study.

1.2 Research Objectives

- Appropriate use of feature engineering to enhance model precision.
- Development of a technique employing machine learning to detect phishing websites
- Compare the proposed models to the previous state of the art work.
- Integration into Django web Framework.

1.3 Statement of the Problem

The spread of Covid-19 has revealed numerous faults and created new opportunities. As a result of Covid-19, [1] found an 81% increase in the number of consumers who utilize e-commerce platforms; this trend has been observed in other African nations as well [1]. However, the danger presented by this pandemic indicates that there has been a 41% increase in phishing-based cyber-attacks [1]. This astronomical surge in cyber-threats demonstrates that attackers are employing the most current methods to breach users. Consequently, an automated system capable of self-improvement and detecting phishing attacks is required.

1.4 Research Questions

- What ML approaches are currently employed to detect phishing assaults in cyberspace?
- What are the challenges of the existing techniques to detect phishing?

1.5 Significance of the Study

We must be able to recognize phishing attacks if we wish to make intelligent judgments and sustain open communication. Since it is not restricted by the assumptions of standard statistical models, an ML model can therefore reveal considerably deeper insights than a human analyst can deduce from data. If models can identify phishing websites that are not recognized by conventional detection methods, the likelihood of high levels of confidence in online transactions increases.

II. Literature Review

This section begins with an explanation of machine learning and moves on to a collection of past research on phishing detection.

2.1 Machine Learning

Machine learning is an application of artificial intelligence (AI) that offers unprogrammed systems with the ability to autonomously learn and improve from experience. This is also the objective of data science. Unlike traditional programming approaches, which rely on predetermined equations, machine learning algorithms train using computational methods. The method grows increasingly accurate as the quantity of samples (data) used to train the computer increases. Focus of machine learning is the development of computer programs that can access data and use it to learn on their own. Machine learning is a subset of artificial intelligence because it allows machines or computers to make decisions based on data without human input or explicit programming. For instance,

intelligent systems based on machine learning algorithms can learn from previous experience or historical data [21].

2.1.1 How Machine Learning Functions:

Using a training data set, a Machine Learning algorithm is taught to generate a model. When the machine learning algorithm receives fresh input data, it generates a prediction based on the model. If the accuracy of the forecast is deemed acceptable, the Machine Learning Algorithm is implemented. If the accuracy is considered insufficient, the Machine Learning Algorithm is retrained with more feature engineering techniques.

2.1.2 Types of Machine Learning

As previously said, Machine Learning is a notion that enables machines to learn from examples and experience without being explicitly programmed. Instead of creating code, you simply feed data to the generic algorithm, and the algorithm/machine constructs the logic depending on the input provided. Supervised learning, unsupervised learning, and reinforcement learning are the different types of machine learning.

2.1.2.1 Supervised Machine Learning

Supervised Learning is the type of learning where the student is supervised by a teacher or instructor. In this type, the dataset serves as an instructor and is responsible for training the model. After the model has been trained, it can begin to make predictions based on any fresh data trained on.

2.1.2.2 Unsupervised Machine learning

The model learns and recognizes data structures through observation. Once a dataset is submitted to the model, it automatically discovers trends and correlations within it by constructing clusters. As a category of apples or mangoes, it cannot label the cluster, but it will distinguish all apples from all mangoes. Suppose we gave the model with photos of apples, bananas, and mangoes; consequently, it would build groupings based on certain patterns and correlations and divide the dataset into these clusters. Now, when the model receives new data, it adds it to one of the generated clusters.

2.1.2.3 Hybrid Machine Learning (HML)

Typically, it is a combination of two distinct machine learning algorithms. For example, a hybrid classification model can be constructed with one unsupervised learner (or cluster) that pre-processes the training data and one supervised learner (or classifier) that learns the clustering result (Fatai Anifowose, 2020). HML is an evolution of the machine learning work flow that incorporates algorithms, methods, or procedures from similar or dissimilar fields of knowledge or application areas in order to complement one another. As there is no one-size-fits-all hat, there is no single machine learning technique that is applicable to all problems. While certain algorithms excel at handling noisy data, they may suffer with large input dimensions. Others may scale well in a high-dimensional input space, but struggle with sparse data. These characteristics provide a solid foundation for utilizing HML to complement the candidate approaches and to compensate for their deficiencies.

2.1 PHISHING ATTACK

A phishing assault is one of the most serious hazards to any firm, and in this part, we outline the study completed on phishing attacks as well as their many varieties. According to the reports of the anti-phishing, working group [5], phishing was first discovered in 1996 when social engineering was used to attack America Online (AOL) accounts. A vast number of people become vulnerable to phishing, particularly those who are unaware of the threats prevalent in the digital environment. According to a Federal Bureau of Investigation (FBI) IC3 report [45], a phishing attack caused $2.3 billion in damages between October 2013 and February 2016. In general, users disregard a website's URL. Occasionally, phishing frauds related with phishing websites can be effectively avoided by evaluating whether a URL corresponds to a phishing website or a legitimate one. A client can avoid the criminal's trap when a website is suspected of being a targeted phish [9].

Conventional methods for detecting phishing attempts are ineffective, detecting just about 20% of phishing attacks. Although machine learning techniques perform well for phishing detection, they are time-consuming, even on small datasets, and cannot be scaled. Using heuristics to identify phishing leads in a large number of false positives. The importance of client awareness for phishing attack resistance cannot be overstated. Phishers utilize bogus URLs to acquire the confidential private information of their victims, such as bank account information, personal information, login, and password. In the following subsection, we examine several works based on the machine learning, hybrid, and deep learning approaches [10].

2.2.1 Machine Learning Based Phishing Detection

Popular machine learning techniques for detecting phishing websites simplify the problem to a straightforward classification issue. The data used to train a machine learning model for a learning-based detection system must incorporate phishing and genuine website categorization characteristics. Various classifiers are employed to identify phishing attempts. Previous research has proven that detection accuracy is

high when robust machine learning algorithms are utilized. A variety of feature selection approaches are employed to reduce the amount of characteristics. The operation of the machine learning model is represented in Figure 1. Training the machine learning model to predict phishing attempts or legal websites requires a set of input data.

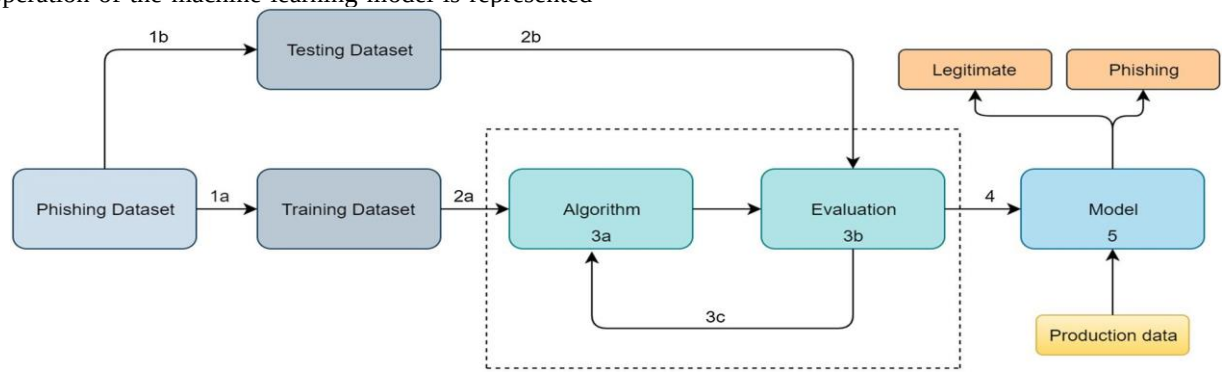

Figure 1 Machine Learning Model [8]

By decreasing the amount of characteristics in a dataset, the visualization becomes more effective and understandable. C4.5, k-NN, and SVM are the most significant classifiers utilized in numerous research and found to be accurate at detecting phishing attacks. Due to the fact that they are built on DTs such as C4.5, these classifiers offer the best level of accuracy and efficiency for detecting phishing assaults [8]. To further explore the identification of phishing assaults, the researchers acknowledged the limitations of their work. Numerous studies highlighted a common limitation: ensemble learning approaches and feature reduction are not utilized in certain trials. [33] employed various classifiers, such as IBK, NB, and SVM. Similarly, [38] employed radio frequency identification to distinguish between phishing assaults and authentic web sites. The writers of [1] utilized Adaptive Neuro-Fuzzy Inference. A resilient system-based strategy that employs integrated characteristics to identify and prevent phishing assaults.

Reference [30] investigated the detection of phishing websites using supervised learning and stacking models. The objective of these research was to improve classification accuracy by employing specific characteristics and stacking the most effective classifiers. Stacking (RF, NN, stowing) beat N1 and N2 classifiers that were also offered. The experiment was conducted utilizing phishing website data sets. The data collection contained 32 pre-processed features corresponding to 11,055 web pages. Alsariera et al (LBET). On phishing website datasets, the proposed meta-algorithms were calibrated and analyzed.

In detecting phishing attempts, the proposed models also outperformed existing machine learning-based algorithms. Therefore, they recommend employing meta-algorithms in the creation of phishing attack identification models. PhishBench is a benchmarking framework proposed by [12] that permits us to evaluate and analyze existing phishing detection features and fully comprehend indistinguishable test conditions, such as a unified framework specification, datasets, classifiers, and performance measurements. Experiments demonstrated a decline in categorization performance when the ratio of phishing to real messages increased from 1 to 10. Execution declined by 5.9% points to 42% for the F1- grade. In addition, PhishBench was utilized to test prior approaches against new and diverse data sets.

Reference [39] suggested a system for intelligently identifying phishing websites. Using proprietary machine learning models, they categorized websites as legitimate or phishing. A precise and intelligent structure for detecting phishing websites was developed using a few classification approaches. The performance of machine learning algorithms was evaluated using the area under the receiver operating characteristic curve (ROC), the F-measure, and the area under the curve (AUC). With a maximum accuracy of 97.61%, Adaboost with SVM outperformed all other classification techniques.

The research by [33] incorporated the Alexa and Phishtank databases. Using four classifiers — NB, DT, KNN, and Support Vector Machine — their proposed method reads each URL sequentially and analyses the host-name URL and path to determine whether a behavior is an attack or legitimate activity (SVM).

The research by [18] develop Enhanced Dynamic Rule Induction, a method for identifying phishing attacks (eDRI). Using feature extraction, the Remove replace feature selection approach (RRFST), and ANOVA, they reduced the number of features. Compared to previous research, their 93.5% accuracy is the highest according to the data. The research conducted by Hota, H, et al. (2018) proposed the Remove Replace Feature Selection Technique for picking features (RRFST). They claim to have stolen the 47-feature phishing email collection from the anti-phishing website maintained by khoonji. The DT was utilized to predict performance measures.

Reference [11] utilized a dataset from the machine learning repository at the University of California, Irvine, which included 2456 unique URL occurrences and 11,055 total URLs, including 6157 phishing websites and 4898 trustworthy websites. They extracted 30 attributes from URLs and used them to predict phishing attacks. There were two possible outcomes: alerting the user that the website is a phishing effort or reassuring the user that the page is secure. They used machine learning algorithms including DT, RF, Gradient Boosting (GBM), General Linear Model (GLM), and Principal Component Analysis (PCA). The research by [10] increased the model's detection coverage using the SMOTE method. They trained models of machine learning including bagging, reinforcement learning, and XGboost. Due to the utilization of XGboost technology, their suggested solution was the most accurate. They analyzed Phishtank's dataset, which consists of 24,471 phishing sites and 3,850 legitimate sites. The authors of [40] used the RF algorithm as a binary classifier and the relief algorithm to select features. Using the dataset from the Mendeley website as input to the feature selection algorithm, they chose effective features. In order to forecast the phishing attempt, they then trained a reinforcement learning system on the selected features.

The research by [24] performed a study on the usage of KNN to detect phishing websites using the Phishtank dataset. He noted that the performance of the suggested model was empirically evaluated and the findings were analyzed. The performed research reveals that the model was successful at detecting phishing assaults. Moreover, he identified the K value, which enhanced the precision of his own phishing assault detection. The proposed model was 85.08% accurate on average. According to the literature, the research community has worked to improve the categorization accuracy of phishing websites by considering a large variety of parameters. However, it appears that less emphasis is placed on the issue of constructing a classification model rapidly without losing accuracy.

2.2.2 Detection of Phishing Attacks Using Hybrid Learning (Hl)

This section compares the HL models employed in state-of-the-art investigations. In this section, we cover the research conducted on different ensembles and how hierarchical learning was utilized to detect phishing assaults. Kumar et al. (2020) removed some irrelevant information from the text and photos and using an SVM-based binary classifier. To categorize authentic and phished emails, they use text parsing, word tokenization, and stop word elimination. The authors of [43] used TF-IDF to select the most significant website elements to include in the search query, but the algorithm was optimized for performance. The proposed strategy was determined to be more precise than existing strategies employing the standard TF-IDF methodology.

The research by [1] proposed a hybrid approach that combines Search and Heuristic Rule and Logistic Regression for efficient phishing attack detection (SHLR). The authors suggested the following three-step process:

- The majority of websites returned in response to a search query are legitimate if the domain name of the web page matches the domain name of the websites returned in response to the query;
- The heuristic criteria given by the character characteristics are legitimate.
- A machine learning model that predicts if the online page is real or an attempt at phishing.

Reference [19] utilized LR, DT, and RF strategies to detect phishing attacks, and they consider the RF technique to be a vastly superior technique. This approach has the ability to identify a small number of false-positive and false-negative results. [42] utilized the phishing dataset from the University of California, Irvine, which included 11,055 samples with 6157 legitimate and 4898 phishing incidents. Combining the KNN and random forest techniques, the EKRV model was utilized. Chiew et al. (2019) utilized two datasets: one with 5000 phishing web pages based on PhishTank URLs and the other with OpenPhish. In addition, 5000 legitimate web pages were generated using Alexa and the URLs of the Common Crawl5 repository. They adopted the Hybrid Ensemble technique.

[34] utilized a dataset from the Website phishing dataset, which is accessible online via a repository at the University of California. There are ten characteristics and 1,353 cases in this dataset. They trained a hybrid RF-SVM model with an accuracy of 94%.

The research conducted by [42] created an ensemble technique based on voting and stacking. They selected the UCI machine learning phishing dataset and retrieved only 23 out of 30 features for additional attack detection. Out of 11,055 instances, the dataset contains 6157 lawful and 4898 phishing occurrences. Utilizing the EKRV model, they foresaw the phishing effort. [19] introduced a hybrid method that combines three techniques: blacklisting and whitelisting, heuristics, and visual similarity.

The proposed solution monitors all traffic traversing the end-machine user's and analyses each URL against a white list of trusted domains. The website analyses a variety of details in order to identify traits. The three outcomes are suspicious websites, phishing websites, and legitimate websites. The machine learning classifier is used to collect and compute scores from data. If the score exceeds the threshold, the URL is automatically marked as phishing and blocked.

Using LR, DT, and RF, they predicted the correctness of their test websites. [41] utilized RF and SVM to detect phishing assaults. They analyzed two types of separate datasets. The first is a collection of thirty features from the machine learning library at the University of

California, Irvine. This dataset comprises 2456 URLs that are either phishing sites or legitimate websites. The second dataset contains 1353 URLs that are identified based on ten attributes and three categories: phishing, non-phishing, and suspicious. [34] analyzed a dataset from a repository at the University of California. On the dataset, they built a hybrid model utilizing RF and SVM, which they employ to predict the accuracy.

2.2.3 Current Practices and Upcoming Obstacles

Phishing attacks continue to be an intriguing approach of enticing an unskilled Internet user into divulging his or her sensitive information to the attackers. Numerous remedies exist, however every time a solution is proposed to counter these attacks, attackers analyze the solution's vulnerabilities and continue their assaults. Numerous solutions for preventing phishing attacks have been proposed in the past. The large increase in COVID-19-related phishing attempts between March 1, 2020, and March 23, 2020, as well as attacks done via online collaboration platforms (ZOOM, Microsoft Teams, etc.), has motivated researchers to dedicate more time to this area. The majority of work, whether at the government or corporate level, educational activities, businesses, or non-commercial activities, has migrated from the on-premises paradigm to the cloud. Increasing numbers of users rely on the Internet for routine chores.

This highlights the importance of having a more accurate and responsive phishing attack detection solution ([26,27, 28]).

Conventional approaches for detecting phishing attempts are ineffective, detecting about 20% of phishing attacks. While machine learning methods produce greater results, they sacrifice scalability and are even time-consuming on small datasets. Using heuristics to identify phishing leads in a large number of false positives. This is a requirement for preventing phishing. In addition to teaching the client on safe browsing, the user interfaces can be modified to include dynamic warnings and, consequently, the ability to identify phishing emails. Despite the fact that IoT devices have access to confidential information, their security architecture and features are in their infancy, making them an exceedingly obvious target for attackers [4].

The entry point for all sorts of malware and ransomware is phishing [13]. Malware attacks against businesses employ ransomware, and ransomware operators demand a substantial ransom in exchange for not disclosing stolen data, a trend that began in 2020. Phishing techniques will spoof COVID-19 and healthcare-related organizations and individuals to exploit unprepared users in 2020. It is preferable to safeguard doors on our end and be proactive in our defenses rather than to contemplate reactive steps to combat phishing attempts after they have occurred.

As a result of imitating the look and functionality of real websites, phishing websites are difficult to identify. Anti-phishing frameworks or browser plug-ins are essential because prevention is superior to treatment. These plugins and frameworks may filter content and identify and block websites suspected of phishing. An organization, such as a bank or government agency, can be notified about phishing attacks at the user's end via an automated reporting component. The time spent remediating a phishing attack can have a negative impact on the productivity and profitability of a firm. In today's world, businesses must provide their employees with the information and tools required to identify and report phishing attempts quickly and proactively, before they cause damage [9].

III. METHODOLOGY

Since its inception, machine learning has shown to be an effective method for revealing hidden patterns in a variety of business contexts. However, the efficacy of the model depends on the application of numerous data pre-processing processes. Data processing is addressed first, followed by feature selection (to assist in selecting the most pertinent feature from the dataset), feature engineering (to normalize the dataset), and model design. "data preparation" refers to the process of preparing (by cleaning and organizing) raw data for use in the construction and training of deep-learning models. This is accomplished by cleaning and organizing the raw data [17]. Attributes of the dataset are inspected for the presence of null values, missing values, and other anomalies at this stage. The dataset comprises 50 features, but after feature selection (Section 3.3), the total number of features utilized in this study is 10. (Table 1).

TABLE 1. DATASET FEATURES AND THE MEANING

NO	COULMN	MEANING
1	NumDash	Length of the website
2	NumHash	Number of hashes
3	NumNumericChars	Token number of numeric characters
4	PctExtHyperlinks	Ext hyperlinks
5	PctExtResourceUrls	External resource urls
6	PctNullSelfRedirectHyperlinks	Length of the redirect hyperlinks
7	FrequentDomainNameMismatch	Frequent domain name mismatch
8	ExtMetaScriptLinkRT	Meta script link RT
9	PctExtNullSelfRedirectHyperlinksRT	Redirect hyperlinks RT
10	CLASS_LABEL	Dependent variable. Phishing (1) or non-phishing (0)

To properly develop the model, the following phases were developed:

3.1 Data Pre-Processing

Even though the phrases "exploratory data analysis" (EDA) and "data pre-processing" are sometimes used interchangeably, research has demonstrated that these two procedures are separate, albeit having considerable similarities. Despite the fact that the names are frequently used interchangeably, this is the case [16]. For a machine learning model to be prepared for good performance, the dataset must be prepared such that it can be utilized by the deep learning model in a straightforward manner. This is essential for the development of a deep learning model [29].

The dataset in Figure 2 is balanced. This shows that, non-phishing websites are of equal size like the phishing websites. As seen in the preceding figure 2, there is no requirement to balance the dataset.

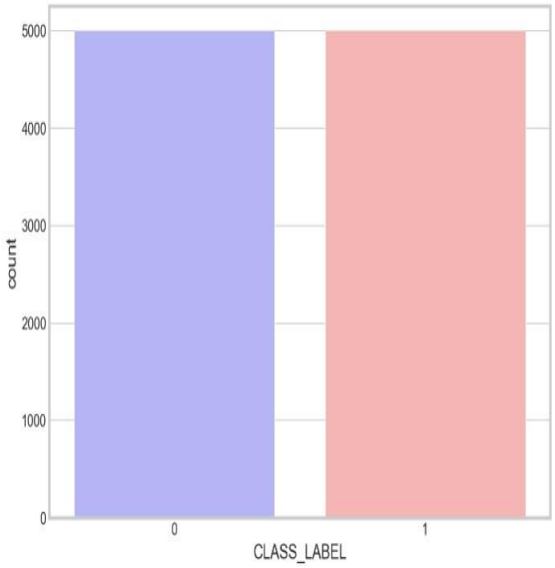

Figure 2 Balanced dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 50 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   id                          10000 non-null  int64
 1   NumDots                     10000 non-null  int64
 2   SubdomainLevel              10000 non-null  int64
 3   PathLevel                   10000 non-null  int64
 4   UrlLength                   10000 non-null  int64
 5   NumDash                     10000 non-null  int64
 6   NumDashInHostname           10000 non-null  int64
 7   AtSymbol                    10000 non-null  int64
 8   TildeSymbol                 10000 non-null  int64
 9   NumUnderscore               10000 non-null  int64
 10  NumPercent                  10000 non-null  int64
 11  NumQueryComponents          10000 non-null  int64
 12  NumAmpersand                10000 non-null  int64
 13  NumHash                     10000 non-null  int64
 14  NumNumericChars             10000 non-null  int64
 15  NoHttps                     10000 non-null  int64
 16  RandomString                10000 non-null  int64
 17  IpAddress                   10000 non-null  int64
 18  DomainInSubdomains          10000 non-null  int64
 19  DomainInPaths               10000 non-null  int64
 20  HttpsInHostname             10000 non-null  int64
 21  HostnameLength              10000 non-null  int64
 22  PathLength                  10000 non-null  int64
 23  QueryLength                 10000 non-null  int64
 24  DoubleSlashInPath           10000 non-null  int64
 25  NumSensitiveWords           10000 non-null  int64
```

Figure 3 Entry Counts

	id	NumDots	SubdomainLevel	PathLevel	UrlLength	NumDash	NumDashInHostname	AtSymbol	TildeSymbol	NumUnderscore	...	IframeOrFrame	MissingT
0	1	3	1	5	72	0	0	0	0	0	...	0	
1	2	3	1	3	144	0	0	0	0	2	...	0	
2	3	3	1	2	58	0	0	0	0	0	...	0	
3	4	3	1	6	79	1	0	0	0	0	...	0	
4	5	3	0	4	46	0	0	0	0	0	...	1	

5 rows × 50 columns

Figure 4 Entries and features of dataset

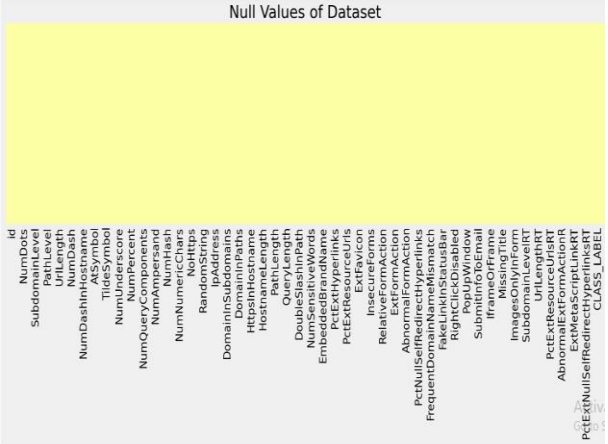

Figure 5 The dataset contains no null value

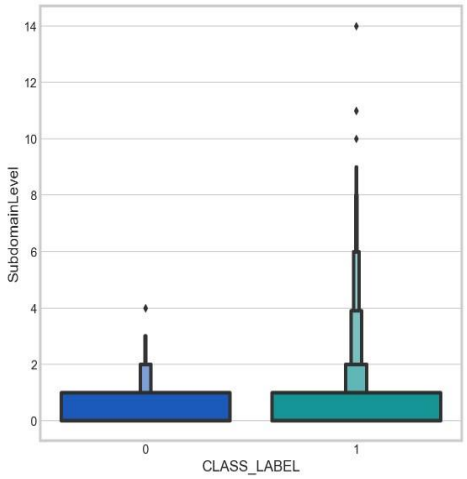

Figure 6 Subdomain Level count with dependent variable

The dataset have 10000 entries and 50 features (columns) as illustrated in figure 3 and figure 4 respectively. Figure 5 shows that the dataset have no null value hence, there is no need to address null value challenge [17]. Checking the effect of one of the independent variable, Figure 6 shows that high number of SubdomainLevel are common in the phishing websites than non-phishing websites.

3.2 FEATURE SELECTION

During the feature selection procedure, the characteristics are whittled down to those that pertain to the dependent variable, either manually or automatically. This step is performed because it has a significant impact on the performance of the model in terms of the time required to construct it and its level of accuracy [23]. Due to the fact that they force the model to learn on irrelevant data, irrelevant dataset characteristics may have a negative influence on training. Due to the fact that irrelevant data acts as noise, poor feature selection will result in unreliable accuracy [23]. When features are picked, there is a reduction in overfitting, an increase in accuracy, and a decrease in the training time required. Through the use of feature selection, it is feasible to reduce the dimensions of a dataset [6].

3.3.1 Manual Feature Selection

As shown in 3.1c, *Id* is one of the features of the dataset. However, it simply represents numbering of each cell of the entries. This does not have any effect on the outcome of the model prediction. This particular feature is quickly removed.

3.3.2 Automatic Feature Selection

During this research, the mutual information categorization method was implemented [18]. By employing this method, one can gain a deeper knowledge of the relationship between the dependent and independent variables. Figure 7 and figure 8 illustrates the result of implementing mutual information tactics.

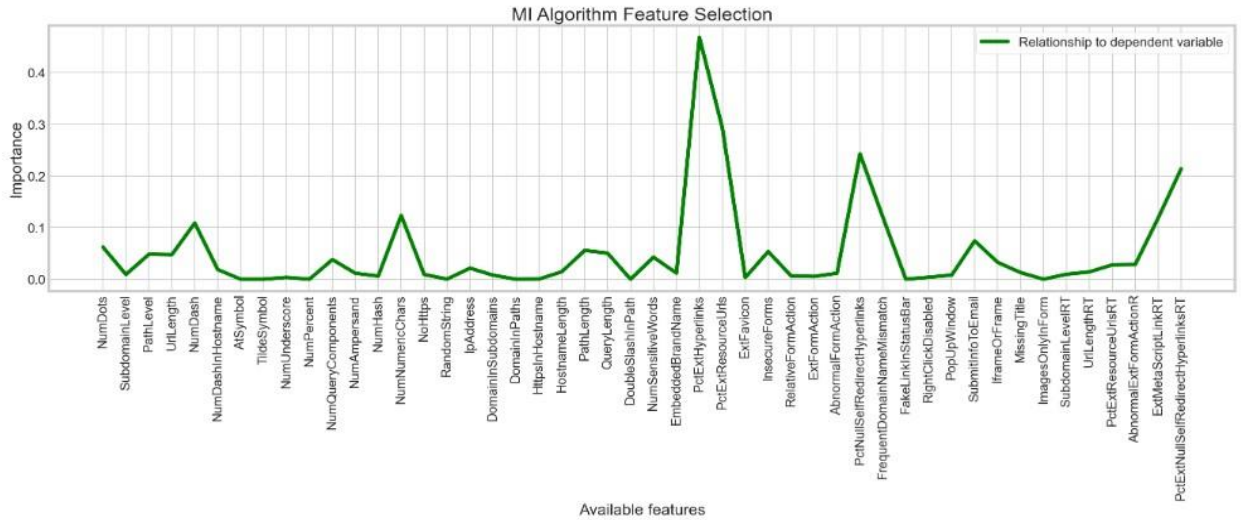

Figure 7 Feature selection (mutual information technique)

From the Figure 7, small importance (0.1 below) to our dependent variable is dropped. The total features used in this research was later dropped to 10 features as shown in Figure 8.

	NumDash	NumHash	NumNumericChars	PctExtHyperlinks	PctExtResourceUrls	PctNullSelfRedirectHyperlinks	FrequentDomainNameMismatch
0	0	0	0	0.000	0.250000	0.0	0
1	0	0	41	0.000	0.000000	0.0	0
2	0	0	0	0.375	1.000000	0.0	0
3	1	0	0	1.000	0.095238	0.0	1
4	0	0	2	1.000	1.000000	0.0	1

Figure 8 Used features after mutual information techniques

3.4 FEATURE ENGINEERING

Since the dataset is balanced, strategies for feature engineering that address the issue of an unbalanced dataset are not covered. In contrast, normalization approaches allow the dataset to be contained inside a specific, closed range, such as [0,1], [-1,1], etc., so that it is equitable and has a greater chance of being successfully predicted. In this study, the interval [-1,1] is used [7].

3.5 MODEL DEVELOPMENT

Since this research utilize the logistic regression (LR), k-nearest neighbor (KNN) and the artificial neural network (ANN). This section documents the important facts about these models. Similarly, the steps required to build Django web application is also documented.

3.5.1 Regression

There are two different forms of regression analysis: linear and logistic. Although logistic regression is used in this study, understanding it requires familiarity with linear regression and the concept of regression itself. Regression is a statistical approach used to model a target value with the aid of independent predictors. This method is mostly employed for forecasting and determining the causal links between variables. Principally distinguishing regression processes are the number of independent variables and the nature of the relationship between independent and dependent variables.

3.5.1.1 Linear Regression

Without linear regression, we cannot apply logistic regression. Simple linear regression is a sort of regression analysis in which there is just one independent variable and a linear relationship between the independent(x) and dependent(y) variables. The following linear equation can be used to depict linear regression.

$$y = a_0 + a_1 * x \quad (1)$$

The purpose of the linear regression technique is to identify the optimal values for a_0 and a_1. Before moving on to the technique, we will examine two essential concepts for a better grasp of linear regression [44].

3.5.1.2 Logistic Regression

Logistic regression is an approach for supervised learning that is applicable when the dependent variable is dichotomous (binary). In contrast to the linear regression outlined previously, real-world problems frequently necessitate non-linear models. Real-world problems can be quadratic, exponential, or logistic. Logistic regression means that potential outcomes are categorical rather than quantitative. Through logistic regression, we may predict categorical outcomes, such as "yes or no will be a phishing website" or "0 or 1 will not be a phishing website." In reality, in the context of this study, decision-making frequently simplifies down to a simple yes or no. In logistic regression, we may make far more fundamental predictions, such as "will this URL at all be a phishing website? For the gradient descent in this study, we employed L2 regularization with SAGA technique.

3.5.2 K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is one of the simplest Machine Learning algorithms for regression and classification challenges [24]. KNN algorithms utilize existing data to classify new data points based on similarity measures (e.g. distance function). Classification is chosen by a vote of the majority of adjacent communities. The data is assigned to the class that is the closest neighbor. Increasing the number of nearest neighbors, or the value of k, can improve accuracy.

3.4.1 Choice of K-Factor

It is challenging to determine the value of k in KNN. The impact of noise on the result will be greater if the k value is small, while the cost of calculating will be higher if the k value is large. If there are two classes, data scientists usually choose an odd number; to choose k, simply use the set $k = sqrt(n)$. However, for this study, we employed the elbow method [35] to get the optimal k factor (figure 9). In this study, we adopted the kd-tree method for the KNN model [36].

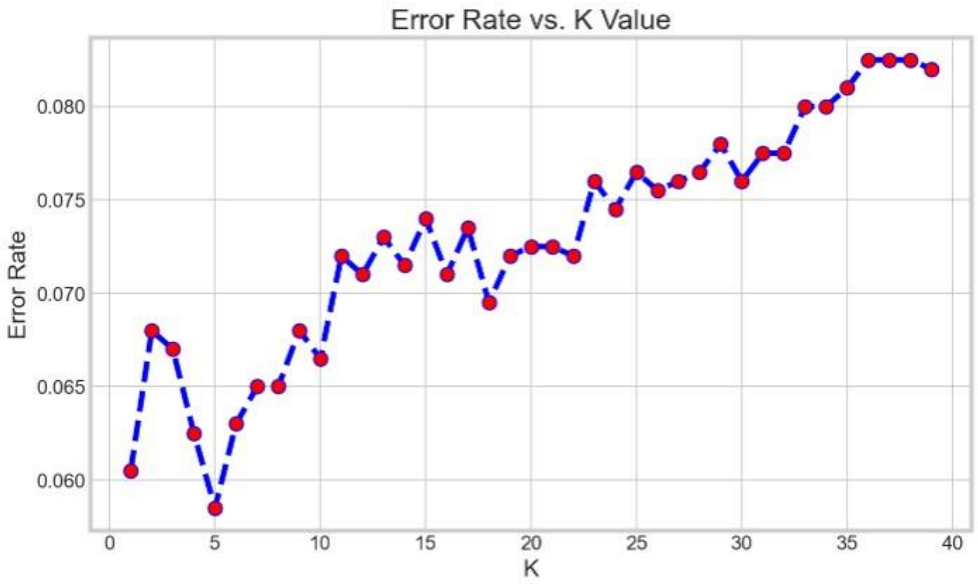

Figure 9 Elbow method to choose our k factor

3.5 Artificial Neural Network (ANN)

Each algorithm in the hierarchy nonlinearly alters its input and produces a statistical model as a result. Iterations will keep going until the results are precise and usable [17]. While feature extraction is a time-consuming process in machine learning, ANN only employs weights to produce the best accurate prediction [17]. The learning rate of this model was accelerated using the Adam optimization algorithms over the course of 400 iterations and an 80-element hidden layer size.

3.6 EVALUATION METRICS

Well-known measuring metrics such as recall, accuracy, f1-score, roc-curve, etc. will be used.

3.6.1 Confusion Matrix

Confusion matrix is used to understand what the model is getting correctly and what it is getting wrongly. Figure 10 shows how confusion matrix generally works then table 5.2 and figure 11 shows the confusion matrix of the model.

True Positive (TP) and True Negative (TN): Real value at the diagonal axis

False Negative (FN): Other values along the horizontal

False Positive (FP): Other value present in the vertical column

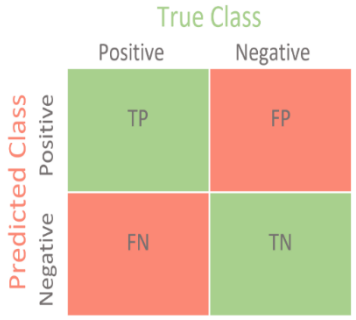

Figure 10 Confusion matrix

3.6.2 Classification Report

This is the recall, precision, accuracy, f1-score, and support as explained below.

a) **Recall**: The TP divided by how many times the classifier predicted that class.

$$\frac{TP}{TP + FN}$$

b) **Precision**: Number of correct predictions divided by how many occurrences of that class were in the test data.

$$\frac{TP}{TP + FP}$$

c) **F1-score**: The weighted harmonic means of the precision and recall values for the test is the F1-score. A high f1-score indicates that the precision is more balanced (Kate Brush, 2021).

$$\frac{2 * Precision * recall}{precision + recall}$$

d) **Support**: The total number of true response samples in the class.

$$TP + FN$$

e) **Accuracy**: Combination of TP and TN divided TP, TN, FP and FN. The higher the better (Kayvan et.al; 2016).

$$\frac{TP + TN}{TP + TN + FP + FN}$$

IV. Implementation Of Practical Work

All of the practical work was carried out using the Python programming language and the frameworks. This section explains the programming language, framework, and integrated development environment (IDE) utilized in this research.

4.1 Programming Language

Python is the most widely used programming language by data scientists. It offers multiple libraries that make machine learning and deep learning development simple. The Python programming language and its associated libraries were heavily utilized in this study.

4.2 Framework/ Libraries

Pandas is used to load, evaluate, and mine data for proper understanding. Moreover, it is used to organize the data such that it is suitable for machine learning and deep learning.

NumPy: This is widely used with pandas to display and do maths on multidimensional arrays.

Scikit-learn: This enables the development of multiple regression, classification, and clustering methods. Additionally, it supports the execution of measurement metrics including classification reports, ROC curves, and confusion scores. matrix. Packages for data visualization with a sophisticated user interface that generate aesthetically pleasing and educational statistics images. Used for 2-D or 3-D array plotting.

Django: Django defines itself as a free and open-source web framework based on the Python programming language and adhering to the model–template–views architectural paradigm.

4.3 DEVELOPMENT ENVIRONMENT (IDE)

IDE is a development environment for software. There are numerous IDEs available for a number of purposes. The popular software among data scientists, Anaconda, is utilized in this inquiry. Small and large software development initiatives can utilize the software.

Jupiter Notebook provides numerous programming languages with interactive computing tools, open standards, and services.

V. RESULT

The model's classifications results is shown in Table 2.

TABLE 2. A CLASSIFICATION REPORT

Model	Precision (%)	Recall (%)	F1-Score (%)	Support (%)	Accuracy (%)
LR	86	86	86	2000	85.60
KNN	94	94	94	2000	94.15
ANN	93	93	93	2000	93.00

TABLE 3. SUMMARY OF THE CONFUSION MATRIX

Models	**TP**	**FP**	**TN**	**FN**
LR	815	164	897	124
KNN	896	83	970	51
ANN	903	76	957	64

The classification results of, logistic regression (LR), k-nearest neighbour (KNN), and artificial neural network (ANN) are shown in Table 5.1. The models' precision, recall, and f1-score are accurate to a degree good percentage, however, KNN seems to have the highest accuracy. Table 5.2 tabulates the confusion matrix presented at figure 5.1. Despite the fact that false positive (FP) is safer than the false negative (FN), since it indicates that the model is suspecting a website to be an phishing website when it is actually a non-phishing website [37], KNN have lowest number of FN which is a very good indication that the model is performing better than the others.

5.1 CONFUSION MATRIX

The model's confusion matrix is depicted in figure 11 to figure 13.

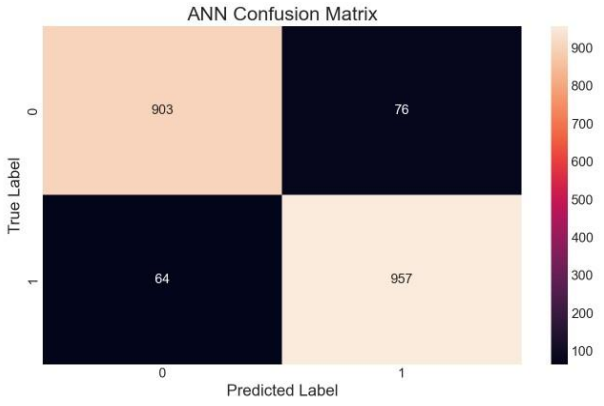

Figure 11 ANN confusion matrix

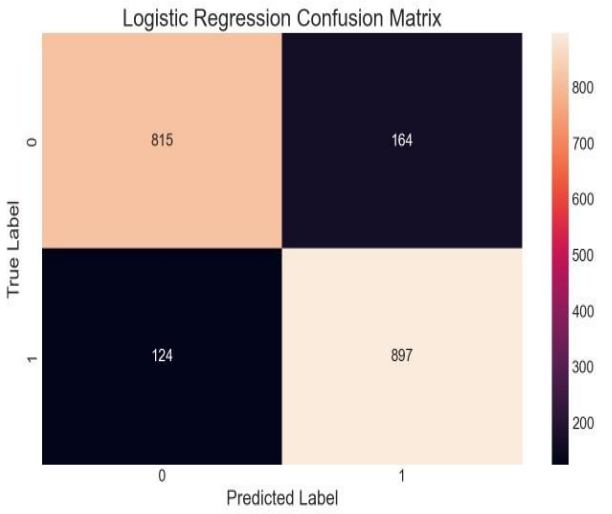

Figure 12 LR confusion matrix

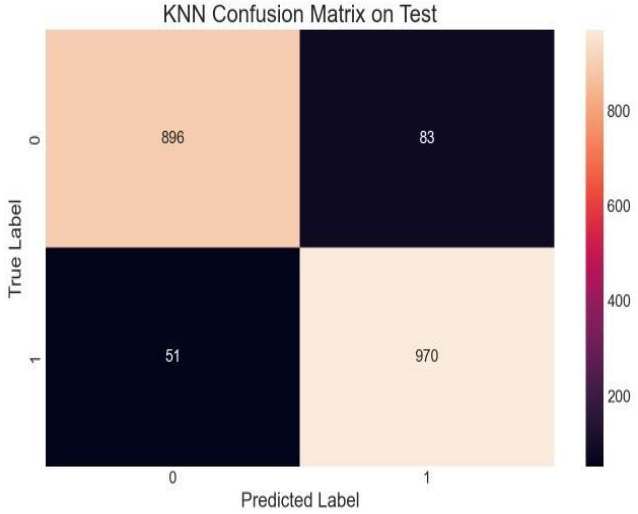

Figure 13 KNN confusion matrix

5.2 *HOW THE PROPOSED MODEL WILL DETECT PHISHING ATTACKS*

To detect the phishing attacks, the proposed model can be integrated as shown in figure 14.

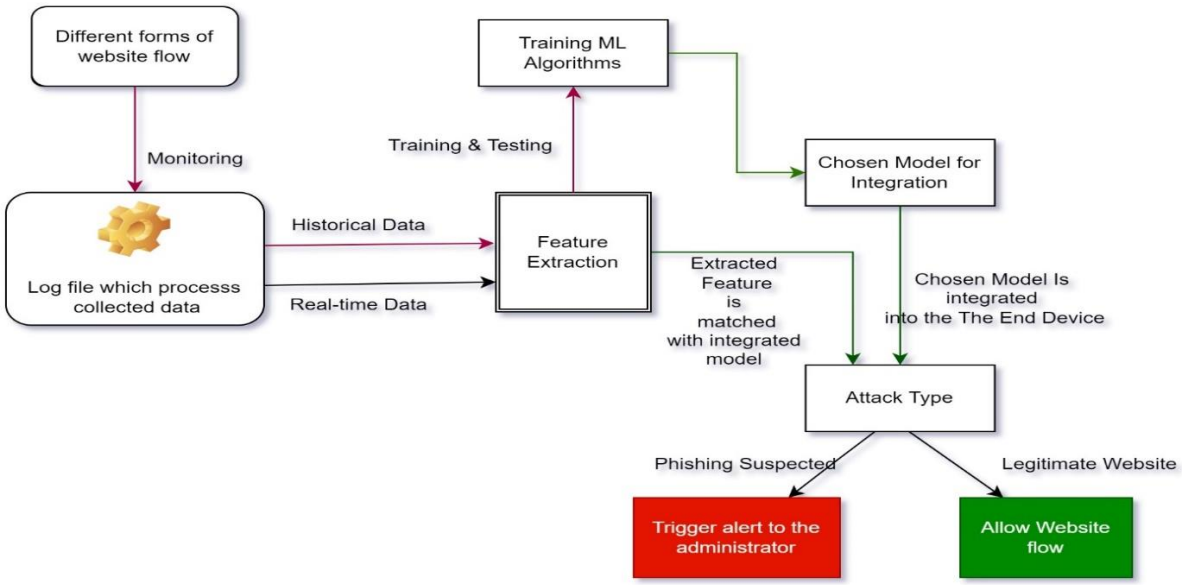

Figure 14 How the proposed model will work

There will be a broad inflow of URLs, as shown in Figure 14, and each URL will be sent for feature selection. This will verify that the features of the URL match the features that will ultimately be used, as shown in Figure 8. A comparison will be made between these features and those utilized by the integrated k-nearest neighbor algorithm (KNN). If it is not a phishing attack, the model makes it possible for the intended request to be delivered when the URL has been determined to be safe and not malicious. On the other hand, access to the URL is denied if it is determined that the effort was a phishing attempt [11].

5.3 COMPARISON WITH OTHER PREVIOUS RESEARCHERS

Reference [18] conducted research on machine learning models for detecting phishing websites. In the course of their study, they investigated a range of models. They utilized a real dataset consisting of 11,000 domains gathered from Phishtank and other sources to determine whether machine learning approaches are more effective than traditional ways for identifying phishing attempts. To achieve the goal, a variety of machine learning (ML) algorithms, such as eDRI,

RIDOR, Bayes Net, SVM, and Boosting, were compared with respect to a number of criteria, such as the predictive model accuracies. This action was taken to achieve the desired outcome. In addition to their effects on the detection rate of phishing attempts, these characteristics were also taken into account as traits. The testing results indicate, with an accuracy of 83%, that the knowledge-based strategy provided by eDRl algorithms appears to be an effective method for avoiding phishing attempts.

The derivation of the characteristics was based on the pooled dataset utilized in this investigation and published in the article authored by [32]. Using ML, they were ultimately able to determine which websites were phishing and which were not. The authors say that they also prioritised the features according to the contribution of each factor used to determine the outcome of a URL link by utilizing Python modules. The purpose of this ranking was to establish how the qualities should be displayed. In order to efficiently detect phishing assaults, the author advised adopting Random Forest (RF), Support Vector Machine (SVM), and Decision Trees (DT), which are all instances of machine learning models. This is due to the fact that the great majority of phishing URLs employ lengthy URLs when used in attacks. By comparing the performance of the models using a confusion matrix to discover which model has the best performance, it is possible to identify which model has the best performance. The scientists concluded that, among all the experiments, the RF had the highest level of accuracy, at 84.81%.

Reference [33] conducted an analysis in which both the Alexa and Phishtank databases played a significant influence. The suggested method checks each URL sequentially and then analyses the host-name URL and path to decide whether a particular behavior is an attack or a lawful action. This review is conducted to determine the legitimacy of a particular behavior. They employ the naive Bayes (NB), the DT, the KNN, and the Support Vector Machine (SVM) as four distinct classifiers. In their study, the authors describe many techniques for identifying phishing websites. Using machine learning algorithms to evaluate and contrast the numerous attributes that real and fraudulent URLs have is one of these ways. They discussed techniques for identifying phishing websites based on the lexical characteristics, host qualities, and page importance characteristics of the websites in question. In order to conduct an evaluation of the characteristics, they studied a variety of data mining techniques. The ultimate goal was to have a deeper understanding of the structure of URLs used to disseminate phishing. The fine-tuned parameters are helpful when picking the machine learning algorithm that will be used to distinguish between phishing and benign websites. The authors of the study were able to attain the highest accuracy feasible for the decision tree, which was 93.78%.

In view of the other research's findings, it is abundantly evident that all the proposed models have a better accuracy than the research proposed by [32]. However, the proposed KNN model have a better accuracy than all the models developed by [33]. This is majorly attributed to the fact that the data preprocessing, feature selection, and feature engineering stages all received a tremendous deal of attention.

5.4 INTEGRATION INTO DJANGO FRAMEWORK

To integrate the proposed model into Django framework, the following steps must be actualized:

- Create a Django project and confirm that the model is running on the assigned server address.
- Create a form which represents all the parameters in the finally used dataset, in this case the number of the finally used independent variables are 9.
- The best trained model, in this case KNN, is saved then loaded in the Django environment.
- When the submit button is clicked, it fires the action which allows the best model (loaded KNN) to compare if the submitted data about a website are phishing or not.

The outcome of the integrated model (figure 15) into Django is shown in Figure 16.

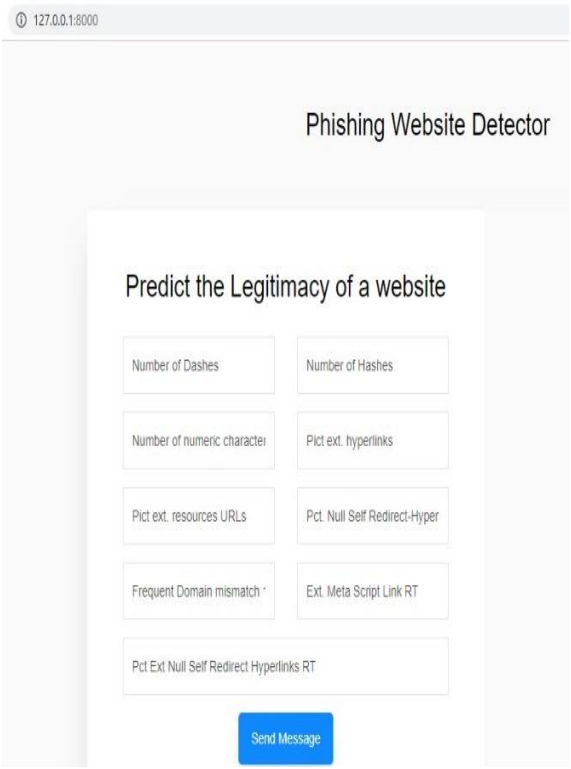

Figure 15 The Integration of the proposed KNN model into Django form

Figure 15 Outcome of the predicted website

VI. Conclusion And Recommendation

In conclusion, ML has gained considerable interest from researchers in a variety of application sectors. This curiosity is growing significantly. ML can automatically extract raw features from complex data without requiring the user to hold any prior knowledge or experience. As a result of the development of new technologies and the exponential growth of data in the era of big data, data mining has emerged as one of the most fascinating subjects in the world of cybersecurity, particularly in the identification of threats. This is especially true in the detection of potential threats. As a result of these findings, a detailed ML model for the identification of phishing has been developed. By analyzing the trends, the study not only provided considerable insight into the existing challenges and barriers that ML has in the detection of phishing attempts, but it also provided this insight. This was made possible by the analysis of trends. Following the completion of data pre-processing, exploratory data analysis (EDA), feature selection, and feature engineering, the suggested model is able to produce more accurate findings than those achieved by prior researchers using the same dataset.

It is recommended that research be conducted on larger datasets, as this can help to utilize many sets of parameters to get the highest potential detection accuracy for future work. In addition, you should prepare to use less fully investigated DL algorithms, such as GAN or DRL, for phishing detection. It is suggested to create heterogeneous ensemble ML models by integrating ML algorithms from other genres, such as CNN-LSTM, DNN-AE, MLP-GRU, etc., in addition to homogeneous structures, in order to examine the efficacy and efficiency of ensemble methods vs individual approaches. This is performed to assess the efficacy and productivity of ensemble techniques. This can be done to examine the efficacy and productivity of ensemble approaches.

The introduction of big data technologies has shown to be crucial for gaining a deeper understanding of a certain organization. It is crucial to train the machine learning (ML) model employed in this study with large datasets in order to uncover hidden characteristics, as greater dataset availability is correlated with improved model performance and real-world credibility.

REFERENCES

[1] Statista. Share of consumers shopping more online since the beginning of the coronavirus (COVID-19) pandemic in selected African countries in 2021. Available at: https://www.statista.com/statistics/1233745/share-of-consumers-shopping-more-online-due-to-covid-19-in-selected-african-countries/

[2] APWG GA, Manning R (2020) APWG Phishing Reports. https://docs.apwg.org/reports/apwg_trends_report_q4_2020.pdf

[3] Almomani A (2018) Fast-flux hunter: a system for filtering online fast-flux botnet. Neural Compute Appl 29(7):483–493

[4] Basit A, Zafar M, Liu X, Javed AR, Jalil Z, Kifayat K (2020) A comprehensive survey of AI-enabled phishing attacks detection techniques. Telecommun Syst 76:139–154.

[5] Rao RS and Pais AR (2019). ‘‘Detection of phishing websites using an efficient feature-based machine learning framework,’’ Neural Compute. Appl., vol. 31, no. 8, pp. 3851–3873. doi: 10.1007/s00521- 017-3305-0.

[6] Aljofey A, Jiang Q, Qu Q, Huang M, and Niyigena JP, ‘‘An effective phishing detection model based on character level convolutional neural network from URL,’’ Electronics:, vol. 9, no. 9, p. 1514, Sep. 2020, doi: 10.3390/electronics9091514.

[7] UNB, 2016. https://www.unb.ca/cic/datasets/url-2016.html

[8] Ahmad R and Alsmadi I, ‘‘Machine learning approaches to IoT security: A systematic literature review,’’ Internet Things, vol. 14, Jun. 2021, Art. no. 100365, doi: 10.1016/j.iot.2021.100365

[9] Huang Y, Q. Yang, J. Qin, and W. Wen, ‘‘Phishing URL detection via CNN and attention-based hierarchical RNN,’’ in Proc. 18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Communications/13th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE), Aug. 2019, pp. 112–119, doi: 10.1109/TrustCom/BigDataSE.2019.00024.

[10] Yang P, G. Zhao, and P. Zeng, ‘‘Phishing website detection based on multidimensional features driven by deep learning,’’ IEEE Access, vol. 7, pp. 15196–15209, 2019, doi: 10.1109/ACCESS.2019.2892066.

[11] Zamir A, Khan HU, Iqbal T, Yousaf N, Aslam F, Anjum A, Hamdani M (2020) Phishing web site detection using diverse machine learning algorithms. Electron Libr 38(1):65–80

[12] El Aassal A, Baki S, Das A, and Verma RM, ‘‘An in-depth benchmarking and evaluation of phishing detection research for security needs,’’ IEEE Access, vol. 8, pp. 22170–22192, 2020, doi: 10.1109/ACCESS.2020.2969780.

[13] Geetha R and T. Thilagam, ‘‘A review on the effectiveness of machine learning and deep learning algorithms for cyber security,’’ Arch. Com- put. Methods Eng., vol. 28, no. 4, pp. 2861–2879, Jun. 2021, doi: 10.1007/s11831-020-09478-2.

[14] Cheng M, Li Q, Wang J, Sun B (2020) LSTM based phishing detection for big email data. IEEE Trans Big Data 8(1):278–288

[15] Mahdavifar S, Ghorbani AA (2019) Application of deep learning to cybersecurity: a survey. Neurocom- putting 347:149–176

[16] Kurtis Pykes (2020). Oversampling and Under sampling A technique for Imbalanced Classification. https://towardsdatascience.com/oversampling-and-undersampling-5e2bbaf56dcf

[17] Kirill Eremenko (2017), Deep Learning A-Z™: Hands-On Artificial Neural Networks, https://www.udemy.com/course/deeplearning/

[18] Neda Abdelhamid, Fadi Thabtah, Hussein Abdel-jaber (2017). Phishing detection: A recent intelligent machine learning comparison based on models content and features. https://ieeexplore.ieee.org/abstract/document/8004877

[19] Patil S and S. Dhage, ''A methodical overview on phishing detection along with an organized way to construct an anti-phishing framework,'' in Proc. 5th Int. Conf. Adv. Compute. Commun. Syst. (ICACCS), Mar. 2019, pp. 588–593, doi: 10.1109/ICACCS.2019.8728356.

[20] APWG | Phishing Activity Trends Reports. Accessed: Apr. 8, 2021. https://apwg.org/trendsreports/

[21] Sullins LL (2006) Phishing for a solution: Domestic and international approaches to decreasing online identity theft. Emory Int'l L Rev 20:397

[22] Sahingoz OK, S. I. Baykal, and D. Bulut, ''Phishing detection from urls by using neural networks,'' in Computer Science & Information Technology (CS&IT). India: AIRCC Publishing Corporation, Dec. 2018, pp. 41–54, doi: 10.5121/csit.2018.81705.

[23] Adebowale MA, Lwin KT, Hossain MA (2020) Intelligent phishing detection scheme using deep learning algorithms. J Enterp Inf Manag. https://doi.org/10.1108/JEIM-01-2020-0036

[24] Tsehay Admassu Assegie (2021), K-Nearest Neighbor Based URL Identification Model for Phishing Attack Detection, https://zenodo.org/record/5499632#.YcvCZWjMK00

[25] Barushka A and Hajek P, ''Spam detection on social networks using cost-sensitive feature selection and ensemble-based regularized deep neural networks,'' Neural Comput. Appl., vol. 32, no. 9, pp. 4239–4257, May 2020, doi: 10.1007/s00521-019-04331-5.

[26] Vayansky I, Kumar S (2018) Phishing–challenges and solutions. Comput Fraud Secur 2018(1):15–20

[27] Wei X, Wei X, Kong X, Lu S, Xing W, Lu W (2020) FMixCutMatch for semi-supervised deep learning. Neural Netw 133:166–176

[28] Wei W, Q. Ke, J. Nowak, M. Korytkowski, R. Scherer, and M. Woźniak, ''Accurate and fast URL phishing detector: A convolutional neural net- work approach,'' Compute. Netw., vol. 178, Sep. 2020, Art. no. 107275, doi: 10.1016/j.comnet.2020.107275.

[29] Chen Z, ''Deep learning for cybersecurity: A review,'' in Proc. Int. Conf. Compute. Data Sci. (CDS), Aug. 2020, pp. 7–18, doi: 10.1109/CDS49703.2020.00009.

[30] Dou Z, Khalil I, Khreishah A, Al-Fuqaha A, and Guizani M, ''Systematization of knowledge (SoK): A systematic review of software-based web phishing detection,'' IEEE Commun. Surveys Tuts., vol. 19, no. 4, pp. 2797–2819, 4th Quart., 2017, doi: 10.1109/COMST.2017.2752087.

[31] Phishtank: https://phishtank.com/

[32] Anshumaan Mishra and Fancy 2021, Efficient Detection of Phishing Hyperlinks using Machine Learning. Available at: https://ijcionline.com/abstract/10221ijci04

[33] James Fancy 2021, Efficient Detection of Phishing Hyperlinks using Machin Learning. Available at: https://ijcionline.com/abstract/10221ijci04

[34] Pandey, A., Gill, N., Nadendla, K. S. P., & Thaseen, I. S. (2018). Identification of phishing attack in websites using random forest-svm hybrid model.

[35] Ankita Banerji (2021), K-Mean: Getting The Optimal Number Of Clusters, accessed from: https://www.analyticsvidhya.com/blog/2021/05/k-mean-getting-the-optimal-number-of-clusters/

[36] Yixi Cai, Wei Xu, Fu Zhang 2021, ikd-Tree: An Incremental K-D Tree for Robotic Applications, https://arxiv.org/abs/2102.10808

[37] Krishna Mridha, Jahid Hasan, Saravanan D and Ankush Ghosh 2021, Phishing URL Classification Analysis Using ANN Algorithm. Available at: https://ieeexplore.ieee.org/abstract/document/9573797

[38] Liew, S. W., Sani, N. F. M., Abdullah, M. T., Yaakob, R., & Sharum, M. Y. (2019). An effective security alert mechanism for real-time phishing tweet detection on twitter. Available at: https://www.sciencedirect.com/science/article/pii/S0167404818309040

[39] Subasi, A., Molah, E., Almkallawi, F., & Chaudhery, T. J. (2017). Intelligent phishing website detection using random forest classi- fier.

[40] Joshi, A., Pattanshetti, P., & Tanuja, R. (2019). Phishing attack detection using feature selection techniques. In International conference on communication and information processing (ICCIP), Nutan College of Engineering and Research.

[41] Jagadeesan, S., Chaturvedi, A., & Kumar, S. (2018). Url phishing analysis using random forest. International Journal of Pure and Applied Mathematics.

[42] Niranjan, A., Haripriya, D., Pooja, R., Sarah, S., Shenoy, P. D., & Venugopal, K. (2019). Ekrv: Ensemble of knn and random commit- tee using voting for efficient classification of phishing.

[43] Jain, A. K., Parashar, S., Katare, P., & Sharma, I. (2020). Phish-skape: A content based approach to escape phishing attacks.

[44] Jason Brownlee (2021), Random Oversampling and Under sampling for Imbalanced Classification, https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/

[45] IC3 2020, https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf