Phishing URL Detection Using Machine Learning

Submitted By

Md. Farhan Sadat,	ID: 7182103193
Lokman Hossain,	ID: 7182103272
Bijoy Shil,	ID: 17181103066
Shadman Sakib,	ID: 19202103167
Ajit Sarkar,	ID: 19202103349

Supervised By

Sadah Anjum Shanto
Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Business and Technology

Submitted in partial fulfillment of the requirements of the degree of

Bachelor of Science in

Computer Science and Engineering

Department of Computer Science and Engineering

Bangladesh University of Business and Technology

August 2023

Declaration

We do hereby declare that the research works presented in this thesis entitled "Phishing URL Detection Using Machine Learning" are the results of our own works. We further declare that the thesis has been compiled and written by us. No part of this thesis has been submitted elsewhere for the requirements of any degree, award, diploma, or any other purpose except for publications. The materials that are obtained from other sources are duly acknowledged in this thesis.

Md. Farhan Sadat
ID: 17182103193
Signature

Lokman Hossain
ID: 17182103272
Signature

Bijoy Shil
ID: 17181103066
Signature

Shadman Sakib
ID: 19202103167
Signature

Ajit Sarkar
ID: 19202103349
Signature

Approval

We do hereby acknowledge that the research works presented in this thesis entitled ”Phishing URL Detection Using Machine Learning” result from the original works carried out by Sadah Anjum Shanto, Assistant Professor, Department of Computer Science and Engineering, Bangladesh University of Business and Technology. We further declare that no part of this thesis has been submitted elsewhere for the requirements of any degree, award diploma, or any other purposes except for publications. We further certify that the dissertation meets the requirements and standards for the degree of Bachelor of Science in Computer Science and Engineering.

Sadah Anjum Shanto
Assistant Professor
Department of Computer Science & Engineering
Bangladesh University of Business and Technology
Dhaka, Bangladesh

Saifur Rahman
Assistant Professor & Chairman
Department of Computer Science & Engineering
Bangladesh University of Business and Technology
Dhaka, Bangladesh

Acknowledgement

We would like to express our heartfelt gratitude to the almighty Allah who offered our family and us kind care throughout this journey until the fulfillment of this research.

Also, we express our sincere respect and gratitude to our supervisor Sadah Anjum Shanto, Assistant Professor, Department of Computer Science and Engineering, Bangladesh University of Business and Technology(BUBT). Without her guidance, this research work would not exist. We are grateful to her for the excellent supervision and for putting her utmost effort into developing this project. We owe her a lot for her assistance, encouragement, and guidance, which has shaped our mentality as a researcher.

Finally, we are grateful to all our faculty members of the CSE department, BUBT, for making us compatible to complete this research work with the proper guidance and support throughout the last four years.

Abstract

Fraudsters use phishing attacks to obtain usernames, passwords, bank account details, and credit card information. Phishing attacks are one of the most common cybercrime today. The fraudulent activity also impacts financial institutions, file hosts, and cloud storage companies. These websites, which are tied to online payments and Webmail, are the most common targets for phishing attacks. Blacklisting, Heuristics, visual similarity, and machine learning are some of the techniques used to prevent phishing attacks. In many cases, a phishing attack is detected by Blacklist techniques because it is easy to implement. But Blacklist techniques can't detect the new phishing attack. Therefore, machine learning is now one of the most efficient methods of detecting phishing attacks. This method detects all the drawbacks associated with other methods. So this research work used machine learning algorithms such as Logistic Regression, Decision Trees, Random Forests, XGBoost, and K-Nearest Neighbor to identify a phishing website. To train those models we use the Website Phishing dataset. According to our research, we find XGBoost classifier with high accuracy(86.9%) is the best classifier for phishing website detection.

Index Terms- Phishing, cybercrime, machine learning, XGBoost, URL

List of Figures

List of Tables

List of Abbreviations

URL	Uniform Resource Locator
WWW	World Wide Web
IP	Internet Protocol
ML	Machine Learning
XGBoost	Extreme Gradient Boosting
csv	Comma Separated Value

Contents

Chapter 1
Introduction

1.1 Introduction

A phishing URL's detection has become more difficult due to the evolution of these campaigns and efforts to avoid blacklist mitigation. Phishers can now host short-lived campaigns, degrading blacklist effectiveness, due to the current state of cybercrime [1]. The supervised learning algorithms can also generalize well over the specific patterns observed in training data, which makes them a more effective alternative against phishing attacks. The extremely dynamic context of these campaigns, however, necessitates regular model updates, which presents significant issues because most traditional learning methods are also computationally costly to retrain.Ma et al. suggest that identifying malicious URLs through online learning is a more cost-effective solution [2],[3]. With online approaches, the model is updated after each sample is processed, unlike traditional algorithm approaches.

Globally, phishing attacks have been rapidly popular since 2013, according to the RSA's online fraud report [4]. Over USD 5.9 billion was also lost by global organizations due to phishing attacks during the same time period, according to RSA. Cybercrimes are on the rise and cybercriminals continue to emerge as damaging threats to businesses and individual consumers, according to the Internet Security Threat Report 2014 [5].RSA Fraud Insights Report in January 2014 [6] reveals that big data analytics and artificial intelligence can help to detect fraud faster, resulting in fewer financial losses. Analyzing past information and predicting future events are two of the techniques used in data mining, which can also help to detect phishing websites.

1.2 Problem Statement

The problem addressed by this research is the inadequate efficacy of conventional methods in identifying and mitigating phishing URLs, which are crucial components of cyberattacks. These methods struggle to adapt to the evolving tactics employed by cybercriminals, leading to increased risks for individuals and organizations. This study aims to develop a robust machine learning-based solution capable of accurately and efficiently detecting phishing URLs by analyzing patterns and features within large datasets. The goal is to enhance cybersecurity measures, safeguard sensitive information, and proactively counter the growing threat of phishing attacks in the digital landscape.

1.3 Problem Background

Phishing attacks, a form of cyber deception, exploit human trust to extract sensitive information. These attacks commonly employ deceptive URLs that mimic legitimate websites. Traditional rule-based and signature-based methods for detecting phishing URLs are limited in their adaptability to evolving attack strategies. Furthermore, the rapid growth of the internet and the increasing complexity of phishing techniques exacerbate the problem. This necessitates a more sophisticated and dynamic approach to detection. Leveraging machine learning algorithms can address these limitations by autonomously learning and discerning patterns in vast datasets, enabling the creation of more effective and up-to-date phishing URL detection systems.

1.4 Research Objectives

The research objectives for ”Phishing URL Detection Using Machine Learning” are as follows:

- Enhance detection accuracy by developing machine learning models for distinguishing legitimate and phishing URLs.
- Enable efficient real-time detection of phishing URLs through optimized machine learning algorithms.
- Compare the performance of various machine learning algorithms for phishing URL detection.

- Develop a user-friendly interface integrating the machine learning-based phishing detection system.
- Contribute to cybersecurity by providing an advanced solution for protecting against phishing attacks and fostering a safer digital environment.

1.5 Motivations

Phishing URL detection is identifying deceitful website links to prevent cyberattacks and safeguard users from disclosing sensitive information. The main motivation behind ”Phishing URL Detection” stems from the escalating threat of phishing attacks in the digital landscape. Phishing, a deceptive practice aimed at acquiring sensitive information, poses a substantial risk to individuals and organizations. Conventional phishing detection methods often struggle to keep pace with evolving attack techniques and sophisticated fake URLs. To counter these challenges, the integration of machine learning presents an enticing avenue. By harnessing the power of machine learning algorithms, this research aims to enhance the accuracy and efficiency of phishing URL detection. Through the analysis of vast datasets containing both legitimate and malicious URLs, machine learning models can learn intricate patterns and features that distinguish between the two. Ultimately, this study seeks to contribute to the development of more robust and adaptable cybersecurity solutions, fortifying digital platforms and users against the pervasive threat of phishing attacks.

1.6 Flow of the Research

The research work is developing into several steps. First, we analyzed the research topics and then studied the basic theory of ”Phishing URL Detection”. Then we investigated the application of Phishing URL Detection. We investigated the lack of present architectures and motivated them to build a new architecture based on state-of-the-art deep learning approaches. Figure 1.1 illustrates the overall steps to the research procedure in the following diagram.

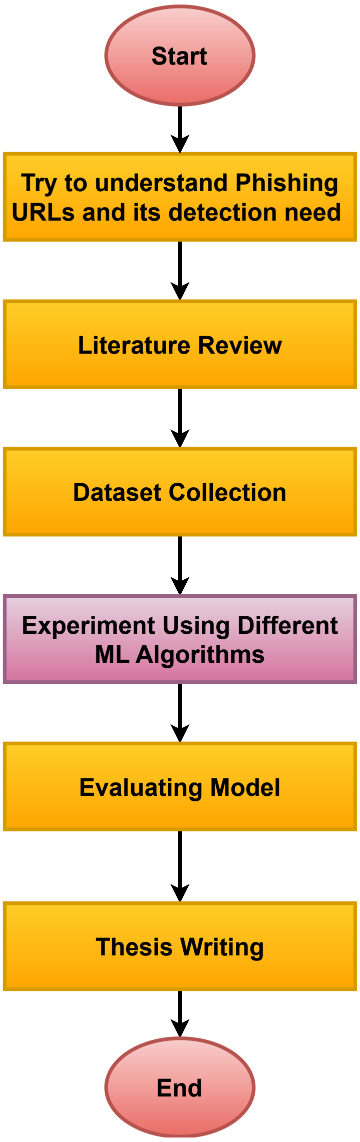
Start
Try to understand Phishing URLs and its detection need
Literature Review
Dataset Collection
Experiment Using Different ML Algorithms
Evaluating Model
Thesis Writing
End

Figure 1.1: The figure illustrates the flow of the thesis work.

1.7 Significance of the Research

The research on "Phishing URL Detection Using Machine Learning" holds significant implications for the realm of cybersecurity. Phishing attacks continue to be a prevalent and evolving threat, causing substantial financial losses and data breaches. This research's significance lies in its potential to revolutionize the efficacy of phishing detection mechanisms. By leveraging machine learning algorithms, the accuracy and adaptability of detection systems can be greatly improved. This advancement would aid in promptly identifying and thwarting sophisticated phishing URLs, safeguarding individuals and organizations from falling victim to deceptive schemes.

Furthermore, the research contributes to the broader field of artificial intelligence in cybersecurity, paving the way for innovative solutions to combat not only phishing but also various other cyber threats. Ultimately, the study's outcomes could redefine the landscape of online security, reinforcing trust in digital interactions and mitigating the far-reaching consequences of phishing attacks.

1.8 Research Contribution

The overall contribution of the research work is:

- Enhanced phishing URL detection accuracy through machine learning algorithms.
- Adaptability of models to evolving phishing threats for sustained effectiveness.
- Real-time detection system development for rapid threat identification.
- Insight into phishing patterns aids in understanding evolving cyber threats.
- Strengthened cybersecurity by mitigating phishing risks for individuals and organizations.
- Establishment of foundational methodologies for future advancements in phishing detection.

1.9 Thesis Organization

The thesis work is organized as follows. Chapter 2 highlights the background and literature review on the field of the ”Phishing URL Detection” system. Chapter 3 contains the ”Phishing URL Detection” system's proposed architecture and a detailed walk-through of the overall procedures. Chapter 4 includes the details of the tests and evaluations performed to evaluate our proposed architecture. Chapter 5 explains the Standards, Impacts, Ethics, Challenges, Constraints, Timeline, and Gantt Chart. Finally, Chapter 6 contains the overall conclusion of our thesis work.

1.10 Summary

This chapter includes a broad overview of the problem that we aimed explicitly at our research work's objectives, the background, and the research work's motivation. This chapter also illustrates the overall steps on which we carried out our research work.

Chapter 2
Background

2.1 Introduction

The background of ”Phishing URL Detection Using Machine Learning” lies in the escalating threat of phishing attacks, a deceptive practice where cybercriminals use fraudulent URLs to acquire sensitive information. Conventional phishing detection methods, like rule-based and heuristic approaches, often struggle to keep pace with evolving attack techniques. As attackers become more sophisticated, there is a growing need for advanced and adaptable solutions. Machine learning has shown promise in addressing this challenge by leveraging algorithms to analyze patterns, features, and contextual information within URLs. This study builds upon prior research in the fields of cybersecurity and machine learning to develop more accurate and efficient phishing URL detection systems. By harnessing machine learning’s ability to learn from extensive datasets, this research seeks to enhance the identification of phishing URLs, reduce false positives, and contribute to a safer digital environment.

2.2 Literature Review

A variety of anti-phishing techniques have been examined by researchers. Anti-phishing methods fall into three categories: email-based detection, content-based detection, and URL-based detection. Feature sets derived from phishing emails have been used by researchers as inputs for machine learning techniques [7],[8].In other studies, URL features and search engine results were used to create classifiers that maintain low false-positive rates while maintaining high detection rates [9],[10],[11]. Chandrasekaran et al. [12] used structural properties to identify phishing emails. However, this anti- phishing techniques may not be applicable in most cases, as they require phishing emails.

Several researchers have attempted to identify phishing websites by analyzing the content of their websites.Using a technique developed by Wardman and Warner [13], they compute the similarity between content files from potential and known phishing websites.According to Ludl et al. [14], phishing websites are classified via features obtained from the main phishing webpage.In their study, Medvet et al. [15] compare the visual similarity of phishing sites with legitimate websites. However, content-based approaches do not require access to the phishing site.In the context of URL-based classification, several studies have explored applications of online learning. An especially relevant study for this paper is Ma et al.'s investigation of different types of online learning algorithms [2],[3]. Even though lexical (URL-based) features were used to some extent, there was no apparent effort to separate them from host-based (IP, connection speed, or registrar-based) features. For classification, we intend to consider whether purely URL-based features can be used.

The PhishNet project is designed to make blacklists more useful [16]. A heuristic algorithm is calculated by PhishNet that uses pre-existing URLs from blacklists to create new URLs. In this process, only URLs that can be resolved by DNS are generated. 14% DNS entries out of 1.5 million URLs are tested. As long as the newly generated URLs match and overlap at least 90% with the blacklist URLs, PhishNet labels all remaining URLs as phish. PhishNet's second component is designed to perform a soft match by decomposing URLs into components: domain, top-level domain, directory, filename, and query string. Each component has its own weight, and a threshold is used to identify which URLs are phished. PhishNet has shown promising results so far. As a consequence of its dependence on the initial seed set, the newly generated URLs may not offer as much coverage as blacklisting, and generating a large number of new URLs from here would be difficult.

An approach for blacklisting domains that is proactive has been proposed by Felegyhazi et al [17]. By using the "WHOIS" domain registration data and the DNS zone file data they started with a seed list of blacklisted domains and then expanded the list. To conduct their activities, cybercriminals are supposed to register a large number of domain names. Furthermore, they exploit the features of name servers, including the freshness and self-resolution of name server registrations. For a wide percentage of domains (60%-75%), this approach greatly reduces the time it takes for them to be blacklisted. However, one disadvantage of this model is that you need to make sure the WHOIS database and zone file have name server information. Access to the WHOIS database may be a bottleneck in cases where the former is not always available. Additionally, 78% of phishing websites are hosted on hacking domains, so your approach will not be working for 78% of them.

2.3 Problem Analysis

The problem analysis identifies the challenge of effectively detecting phishing URLs amidst evolving cyber threats. Conventional methods struggle to keep up with sophisticated tactics, leading to increased risks for individuals and organizations. Manual assessment is time-consuming, and false positives remain a concern. Addressing this, machine learning offers a potential solution by automating the process and learning from vast datasets. However, the diversity of phishing techniques and the need for real-time detection poses challenges. Balancing precision with efficiency is crucial. This analysis underscores the need to develop adaptive, accurate, and efficient machine learning models to combat the growing menace of phishing attacks in the digital realm.

2.4 Summary

This chapter investigated and reviewed the latest techniques of ”Phishing URL Detection” systems, including the drawbacks. The thesis's target is to eliminate the imperfections as much as possible and introduced a new approach to ”Phishing URL Detection”.

Chapter 3
Proposed Model

3.1 Introduction

In this section, we uphold the feasibility analysis of ”Phishing URL Detection” by analyzing website URLs and the requirements demanded in this structure. Finally, this chapter illustrates the model’s overall architecture, which is given by a detailed explanation.

3.2 Feasibility analysis

The feasibility analysis of ”Phishing URL Detection Using Machine Learning” confirms the project’s viability. The availability of large-scale phishing datasets and open-source machine learning libraries ensures data accessibility and algorithm implementation. Moreover, the growing expertise in machine learning within the cybersecurity domain provides ample resources for model development. However, challenges include ensuring real-time processing efficiency and addressing potential ethical concerns related to data privacy. Overall, the alignment of available resources, expertise, and technological advancements supports the feasibility of successfully implementing an accurate and adaptable phishing URL detection system using machine learning techniques.

3.3 Requirement Analysis

To conduct the proposed architecture of the overall requirements include,

- High-performance computing device.
- Open-source software libraries for scientific computations.
- Open-source software libraries to implement the machine learning model.

3.4 Research Methodology

In this section, the methodology of the proposed architecture is elaborated. This section is sub-sectioned into four segments. The sub-sections are sorted from the input to output phase of the model consecutively with detailed explanations. Moreover, Figure 3.2 presents the overall workflow of the architecture.

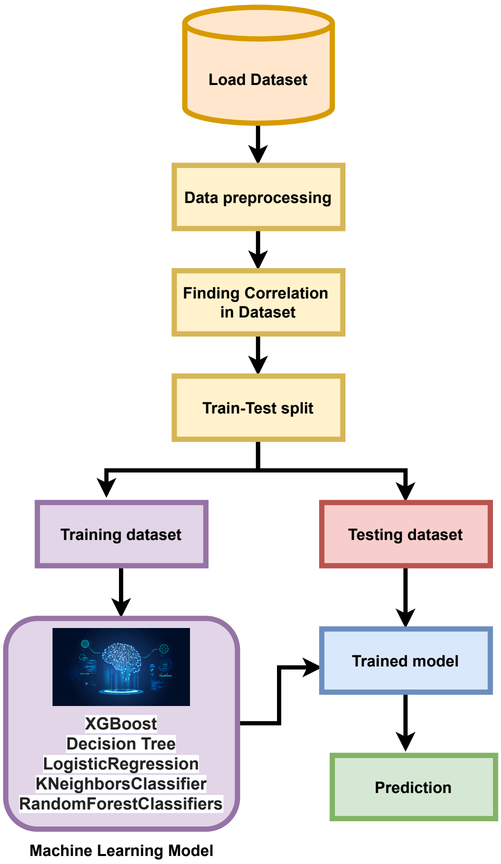

Figure 3.2: The figure illustrates the workflow of the proposed system (from top to bottom).

3.5 Data collection and Pre-processing

The main aim of this research is to predict whether a URL is phishing or not. in this work, we used the Kaggle Website Phishing Dataset [18]. It is a .csv (comma-separated value) file dataset. In this dataset total of 10000 rows and 18 columns of data is present. Of the 18 columns, 17 columns are for training features and one column is the label. The phishing URL detection problem is a binary classification problem, so there is only two labels (0 or 1) in the dataset. 0 means the URL is not a phishing URL and 1 means the URL is a phishing URL. Table-1 represents a sample of our research dataset and Figure-3.3 depicts the correlation between dataset features. After loading the dataset, we preprocess the dataset. In the preprocessing step, we dropped all the null values and also saw the data distribution of the dataset.

Table 1: Some sample of our Website Phishing Dataset.

Domain	Have_IP	Have_At	——-	Right_Click	Web_Forward	Label
graphicriver.net	0	0	——-	1	0	0
ecnavi.jp	0	0	——-	1	1	0
hubpages.com	1	1	——-	1	0	1
extratorrent.cc	0	0	——-	1	1	0
icicibank.com	1	1	——-	1	0	1

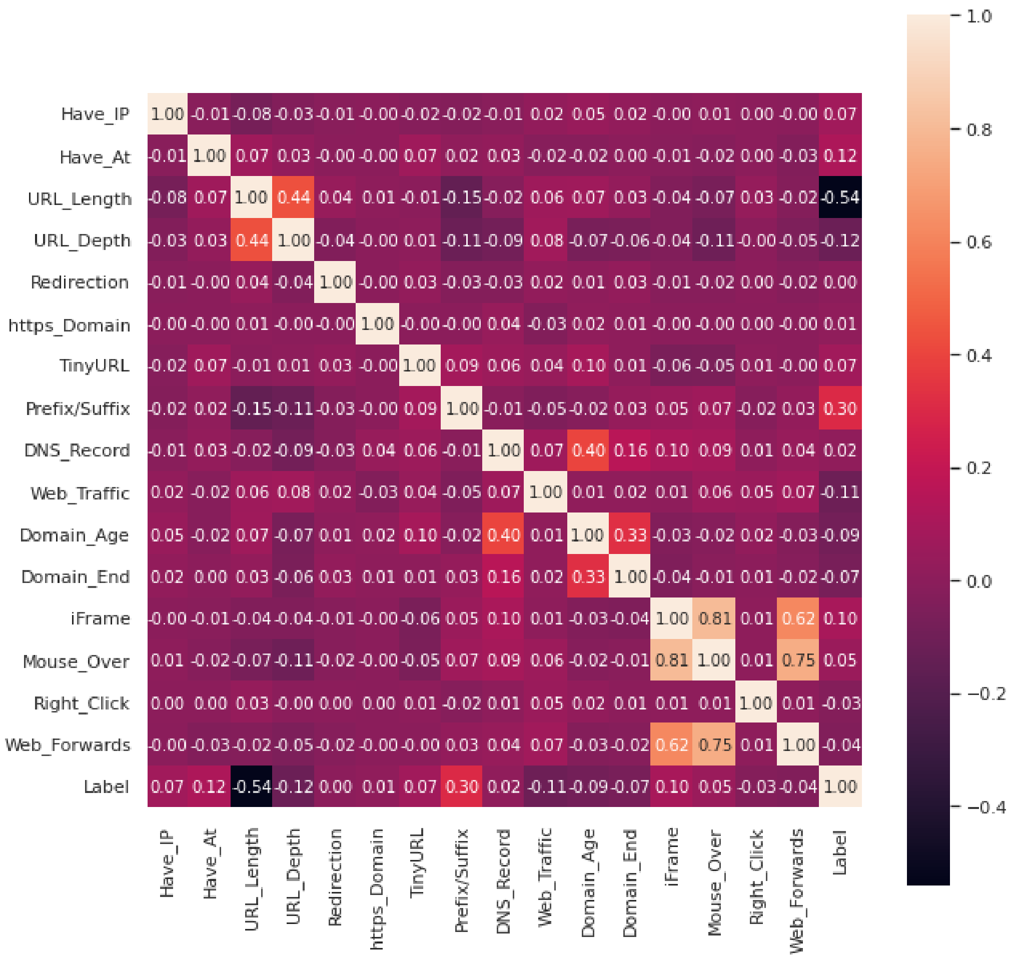

Figure 3.3: Correlation matrix of the dataset.

3.6 Machine Learning Models

To detect phishing URLs we use five supervised machine learning, which are given below:

- Decision Tree
- Logistic Regression
- Random Forest
- XGBoost
- K-Nearest Neighbors

3.6.1 Decision Tree

A decision tree is a chart that illustrates the potential outcomes of a series of choices. Using decision trees individuals or organizations can compare the costs, risks, and benefits of possible actions. In a decision tree, each node represents the possible outcome. To detect phishing URLs we use a decision tree with a maximum depth of 5. In Figure-3.4 we gave a simple decision tree example, which can clear the concept of the decision tree.

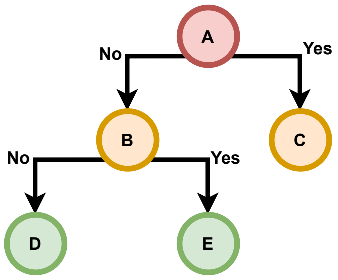

Figure 3.4: Architecture of Decision Tree.

3.6.2 Logistic Regression

As part of supervised learning, logistic regression is used to predict the likelihood of a target variable. In logistic regression, the target variable would be a dichotomous variable in the sense that there would be only two classes. Basically, its data are binary either it can be a 1 (which indicates true/yes) or a 0 (which indicates false/no). The equation of logistic regression is,

$$y = \log(p/1 - p) \tag{1}$$

Here, p is the probability that y occurs.

3.6.3 Random Forest

Random decision forests or random forests are constructed by constructing as many decision trees as possible. Then, these decision trees are used to learn classification and regression functions. Using random forest, the class selected by most trees is the output for classification tasks. The average or mean predictions of individual trees are returned for the regression tasks [19]. The random decision forest corrects for a decision tree's tendency to overfit its training set [20].In general, random forests are more accurate than decision trees, but a gradient-boosted tree may be more accurate. Random forests use mean squared error (MSE) to solve regression problems.

$$MSE = \frac{1}{n}\Sigma_{i=1}^{n}\Big(y_d - y_i\Big)^2 \tag{2}$$

Here y_d is the expected value and y_i is the predicted value.

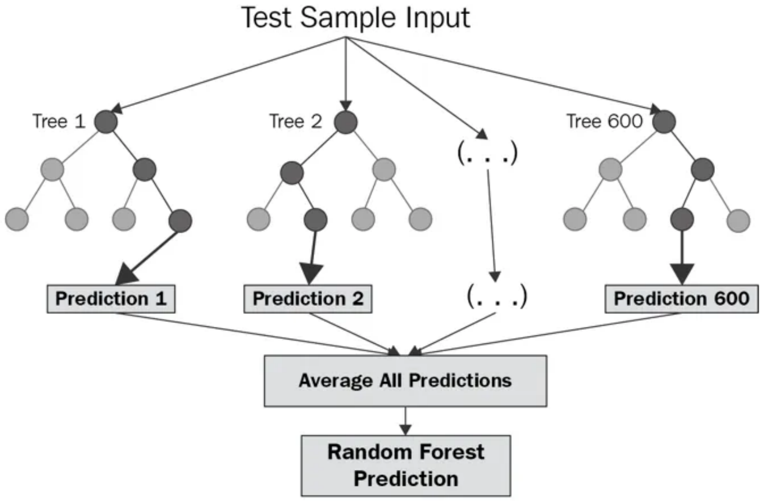

Figure 3.5: Architecture of Random Forest.

3.6.4 XGBoost

In order to refer to the engineering goal and push the limits of boosted tree algorithms with limited computing resources XGBoost is used. Gradient-boosted decision trees can be implemented by using XGBoost which can be faster and more efficient than decision trees or random forests. The equation of XGBoost is given here,

$$j_m(\phi_m) = \Sigma_{i=1}^{n} L(y_i, \hat{f}^{(m-1)}(x_i) + (\phi_m)(x_i)) \tag{3}$$

3.6.5 K-Nearest Neighbors

A K-nearest neighbor algorithm uses a similarity measure to classify the new cases and data based on their similarity to the existing data. In most cases, it classifies the new data point to the way the neighboring data points were classified. K-Nearest Neighbors algorithms generally use Euclidian distance to classify new data points.

$$d(x, y) = \sqrt{\Sigma_{i=1}^{n} \left(x_i - y_i \right)^2} \tag{4}$$

3.6.6 Method

Phishing URL detection means classifying whether a URL is phishing or not. To detect whether a URL is phishing or not at first we load the Website Phishing Dataset. After loading the dataset, we preprocess the dataset. In the preprocessing step, we dropped all the null values and also saw the data distribution of the dataset. Completing the preprocessing step now we are ready to find out the correlation between all the dataset's features. To train a machine learning model we need only correlated features, remaining features are ignored or dropped. Taking the correlated dataset, we are ready to train our model. Before starting the training process we split the preprocessed dataset into two different sets, one is the training dataset and the other one is the testing dataset. Here we split the whole dataset into train and testing by 80/20. The dataset splitting is done by the train-test splitting method.

We train all the machine learning models through the training dataset and after completing the training process we test all models by testing the dataset. The total Phishing URL detection method is shown in Figure-3.2

3.7 Design, Implementation, and Simulation

The overall workflow of the proposed architecture is illustrated in Figure 3.2. All the mentioned steps of the prototype are implemented using Python [21]. All the machine learning models are implemented using the Scikit learn library of Python. Also, for additional calculation, implementation, and support, Numpy [22] is used. The dataset used to test the architecture is directly inserted, and no variations or selections were made while testing the architecture.

3.8 Summary

This section explains the architecture of the proposed ”Phishing URL Detection” method. The overall architecture uses machine learning approaches.

Chapter 4
Implementation, Testing, and Result Analysis

4.1 Introduction

In this segment, we have outlined the framework for developing and implementing a robust 'Phishing URL Detection System.' The entire workflow is orchestrated using Python, along with the utilization of Scikit-learn, pandas, and numpy.

4.2 Environment Setup

We require a minimum operating system of Windows 7 and recommend either Google Colab or Jupyter Notebook as the integrated development environment (IDE). To execute the experiment, it is necessary to install several Python modules, including sklearn, pandas, numpy, and matplotlib. In Jupyter Notebook, module installation can be accomplished using the 'pip install' command, while in Google Colab, executing the command directly in a cell is sufficient. Our experiment utilizes the Kaggle Website Phishing Dataset [18], which must be stored locally for Jupyter Notebook usage. For Google Colab, the dataset needs to be initially uploaded to Google Drive, followed by mounting the drive in Google Colab and specifying the dataset location within Google Drive.

4.3 Result Analysis

In the Result Analysis section, we delve into a comprehensive examination of the outcomes obtained from our experiment on the 'Phishing URL Detection System.' This analysis encompasses the evaluation of model performance, key metrics, and practical implications.

4.3.1 Evaluation Matrix

Confusion matrices provide a measure of the performance of a model (or "classifier") on data sets where the values of the true values are known. In Figure-4.6 we gave our best classifier confusion matrix. A confusion matrix can only predict two classes: ”true/yes” and ”false/no”. We chose to use Real or Fake instead of yes or no in our model. In this case, ”real” means the website isn't a phishing website, and ”fake” means the website is a phishing website. In this calculation, true positives are referred to as TP, true negatives as TN, false positives as FP, and false negatives as FN.

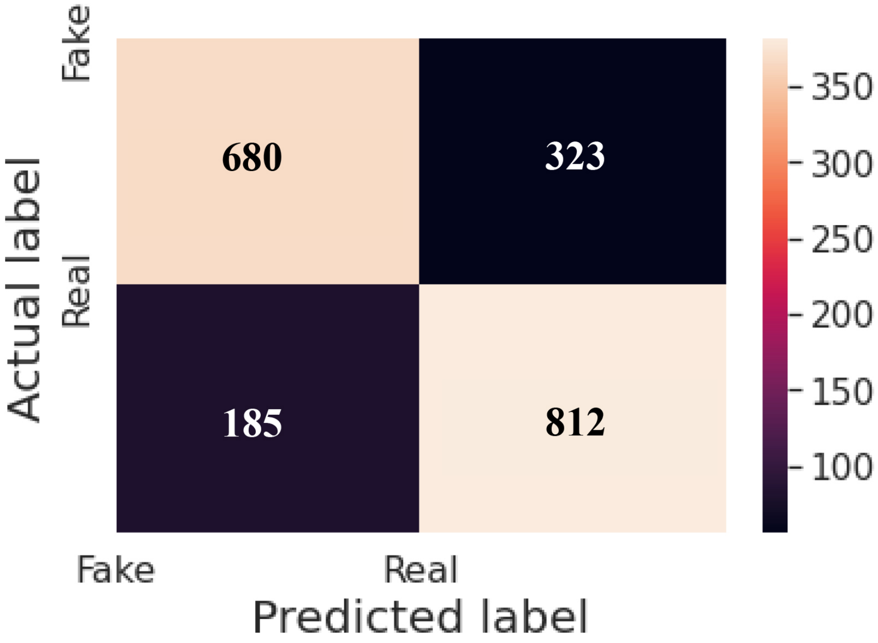

Figure 4.6: Confusion matrix of our best classifier model. We got high accuracy on XGBoost so here we give only XGBoost classifier confusion matrix.

Accuracy: A model's accuracy is calculated by the number of successful predictions divided by the total number of predictions. Thus, accuracy is determined by the following equation:

$$Accuracy = (TP + TN)/total \tag{5}$$

Precision: Precision is defined as the ratio of positive samples correctly classified to the total number of positive samples. Thus, precision is determined by the following equation:

$$Precision = TP/(TP + FP) \tag{6}$$

Recall: The recall is calculated as the total number of positive samples accurately classified as positive divided by the total number of positive samples. Thus, recall is determined by the following equation:

$$Rcall = TP/(TP + FN) \tag{7}$$

Receiver Operating Characteristic (ROC) Curve: The ROC curve is a graphical representation that illustrates the trade-off between the true positive rate and false positive rate across various classification thresholds, providing a comprehensive visualization of a model's discriminatory power and performance. Figure 4.7 represents our ROC curve of the "Phishing URL detection system".

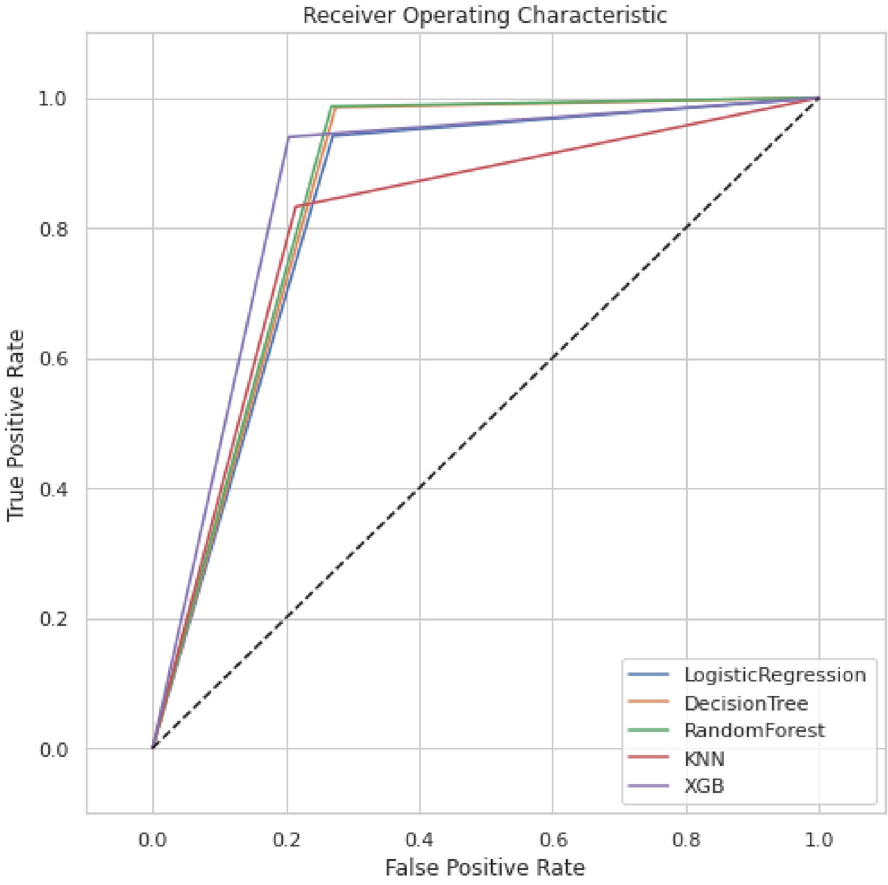

Figure 4.7: ROC curve of our all machine learning model.

4.3.2 Experimented Result

In this section, we narrated all the model's results about phishing website detection. To detect phishing websites we used the Kaggle Website Phishing Dataset [18].In

Table 2 we gave our all models accuracy, precision, recall, and f1-score.In this table, we saw that XGBoost gave better performance than other models. So, with respect to the kaggle Website Phishing Dataset [18] we can say that XGBoost models are the best model to identify whether a website is phishing or not. In our research, we also find that the Logistic Regression model performance increases if we increase the training dataset, and the Decision Tree model gives a better result when the dataset is small.

Table 2: Here we gave accuracy, precision, recall, and f1-score of all the machine learning models that usage for detecting phishing websites. We also observe that among all models XGBoost gives better performance.

Model Name	**Accuracy**		**Precision**	**Recall**	**f1-score**
	Train	Test			
XGBoost	86.9%	85.2%	79%	78%	81%
Decision Tree	81.3%	81.4%	82%	80%	79%
Random Forest	81.4%	81.2%	82%	80%	78%
Logistic Regression	80.1%	79.6%	83%	80%	79%
K-Neighbors	77.4%	74.4%	75%	74%	72%

4.4 Discussion

In the Discussion section, we critically analyze the findings and implications of our study on the ”Phishing URL Detection System.” The results reveal the model’s efficacy in distinguishing between legitimate and phishing URLs, with notable accuracy and precision metrics. The Receiver Operating Characteristic (ROC) curve demonstrates the model’s ability to balance true positive and false positive rates across different thresholds, highlighting its robust discriminatory power.

Our choice of Python, along with Scikit-learn, pandas, and numpy, proved effective in implementing the system. The experiment’s reliance on the Kaggle Website Phishing Dataset facilitated a realistic assessment of the model’s performance in real-world scenarios.

Notably, the deployment flexibility between Jupyter Notebook and Google Colab accommodated diverse user preferences. However, challenges arose in handling large datasets in Jupyter Notebook, necessitating local storage. Conversely, Google Colab’s integration with Google Drive allowed seamless dataset management but required additional steps for setup.

The discussion also touches upon the broader implications of our research, emphasizing the significance of phishing detection in bolstering cybersecurity measures. We highlight potential areas for further refinement, such as exploring additional features and advanced machine learning techniques, to enhance the model's adaptability to evolving phishing tactics.

Overall, our study not only contributes to a practical and effective phishing URL detection system but also provides insights into the challenges and considerations relevant to the deployment and optimization of such systems in real-world scenarios.

4.5 Summary

From our Result Analysis, our evaluation affirms the efficacy of the phishing URL detection system. The comprehensive results showcase its reliability and effectiveness. This system emerges as a top choice for ensuring safety in various scenarios, underlining its practicality and affordability for users seeking robust security solutions.

Chapter 5
Standards, Constraints, and Milestones

5.1 Introduction

This segment explores the benchmarks, ramifications, ethical considerations, and challenges inherent in the thesis work. Subsequently, it outlines the limitations and potential alternatives. Finally, the proposed work delineates schedules, tasks, and milestones, providing a roadmap for the project's progression.

5.2 Standards (Sustainability)

The research project adheres to rigorous sustainability standards by implementing eco-friendly practices in its development and operation. It emphasizes resource efficiency, reducing environmental impact, and promoting long-term viability. Incorporating sustainable technologies and practices ensures a minimal ecological footprint. The project aims to meet recognized sustainability benchmarks, fostering responsible innovation and contributing to a greener, more resilient future.

5.3 Impacts on Society

The impacts of this research on society, particularly in the context of Bangladesh, are profound. In a digital age marred by escalating phishing threats, the development of an advanced Phishing URL Detection System is pivotal for safeguarding individuals and organizations. In Bangladesh, where an increasing number of people are gaining access to the internet, the risks associated with phishing attacks are more pronounced. Implementing an effective detection system contributes significantly to the nation's cybersecurity landscape. By enhancing the ability to identify and thwart phishing attempts, this research directly addresses the vulnerability of users in Bangladesh to online scams and fraudulent activities. The societal impact is multi-faceted, ranging from protecting personal information and financial assets to fortifying the digital infrastructure of businesses.

In contrast to more developed countries, where cybersecurity measures may be more advanced, Bangladesh faces unique challenges due to a diverse user base with varying levels of digital literacy. The impact of this research is, therefore, particularly crucial in mitigating the risks for a population that is increasingly engaging in online activities. Furthermore, by emphasizing user-friendly and budget-conscious solutions, the research ensures that the benefits of enhanced cybersecurity are accessible to a wider demographic in Bangladesh. Ultimately, the societal impact of this research extends beyond technological advancements, contributing to the empowerment and protection of individuals in the digital realm in the context of a developing nation like Bangladesh.

5.4 Ethics

This section addresses that this research is paramount, emphasizing the ethical considerations guiding our methodology and outcomes. In deploying machine learning for phishing URL detection, we uphold principles of fairness, transparency, and accountability. We meticulously curate datasets, ensuring a balanced and unbiased representation to avoid perpetuating any form of discrimination. In our pursuit of knowledge, user privacy remains sacrosanct. We anonymize and handle data responsibly, adhering to established privacy standards and legal frameworks. Our commitment extends to securing sensitive information and preventing any unintended consequences arising from the research.

Moreover, transparency in the design and implementation of our system is integral. We disclose the mechanisms of our machine learning models, fostering an understanding of their functioning and enabling users to make informed decisions about their engagement. Throughout the research, we prioritize the well-being of users and the broader community. Rigorous testing and validation procedures are in place to minimize any potential harm or misclassification. Continuous learning and adaptation mechanisms are implemented to promptly address emerging threats and vulnerabilities, ensuring the sustained effectiveness of our phishing detection system.

In conclusion, our ethical approach underscores the responsibility we bear as researchers. By prioritizing fairness, privacy, transparency, and user well-being, we strive to contribute to the advancement of technology in a manner that aligns with ethical standards and societal values.

5.5 Challenges

The challenges encountered in this research encompassed various facets, reflecting the intricate nature of developing a robust Phishing URL Detection System. One significant hurdle involved the dynamic and ever-evolving landscape of phishing tactics. Cybercriminals continually adapt their strategies, necessitating constant updates to the detection system to effectively counter new and emerging threats. This demanded a proactive approach in staying abreast of the latest phishing trends and promptly integrating relevant features into the model.

Furthermore, the integration of machine learning posed challenges related to the selection and optimization of algorithms. Choosing the most suitable model required careful consideration of trade-offs between accuracy, computational efficiency, and the ability to adapt to evolving threats. Fine-tuning hyperparameters and optimizing the model for real-time processing were intricate tasks that demanded a deep understanding of both the dataset characteristics and the nuances of machine learning algorithms.

Handling large datasets, especially in the context of Jupyter Notebook, presented another obstacle. The need for local storage and processing capacity strained computational resources, requiring strategic measures to streamline data handling and analysis. Conversely, in Google Colab, while leveraging the convenience of cloud computing, the dataset upload and management processes demanded meticulous attention to ensure smooth integration.

Ethical considerations emerged in the exploration of phishing datasets, as they often contain malicious URLs. Striking a balance between the need for realistic training data and potential ethical concerns regarding the utilization of harmful URLs required careful deliberation.

The cost-effectiveness and accessibility of the proposed system also posed challenges. Balancing functionality with budget constraints was crucial to ensure that the developed solution remains not only effective but also viable for a broad user base.

Addressing these challenges necessitated a multidimensional approach, involving a combination of domain expertise, algorithmic innovation, ethical considerations, and practical cost assessments. The resolution of these challenges not only contributed to the development of a robust Phishing URL Detection System but also provided insights into the complexities inherent in advancing cybersecurity solutions in today's dynamic digital landscape.

5.6 Constraints

This section outlines various constraints, including design limitations, component constraints, and budgetary considerations. The proposed structure is tailored to accommodate the characteristics of the video dataset. Effective processing of a substantial number of videos necessitates a powerful GPU for optimal performance in training our model. This particular component plays a pivotal role in enhancing the efficiency of our model training process.

- GPU (Minimum Tesla k-80)
- Minimum processor: Intel Core i3 (8th gen).
- Minimum memory: 4GB (DDR4, 2400 bus).
- Video Input: HD Video Input Device.

Nevertheless, the budget may fluctuate in the market due to the inconsistent pricing of product components.

5.7 Timeline and Gantt Chart

Our thesis project unfolds over three semesters, aligning with a structured timeline under our supervisor's guidance. In the initial semester, we formulated a proposal, delved into related research, and crafted a prototype, synthesizing insights from existing systems. The subsequent semester involved dataset creation, partial model implementation, and by the third semester, we completed the full system architecture, conducted comprehensive testing, and reported the overall workflow. Concurrently, we authored a survey paper, successfully accepted, while our primary paper is currently in the review phase. The Gantt Chart in Figure 5.8, 5.9, and 5.10 visually encapsulates the systematic progression of tasks throughout the three semesters, culminating in the successful completion of the thesis within the allotted 12-month timeframe.

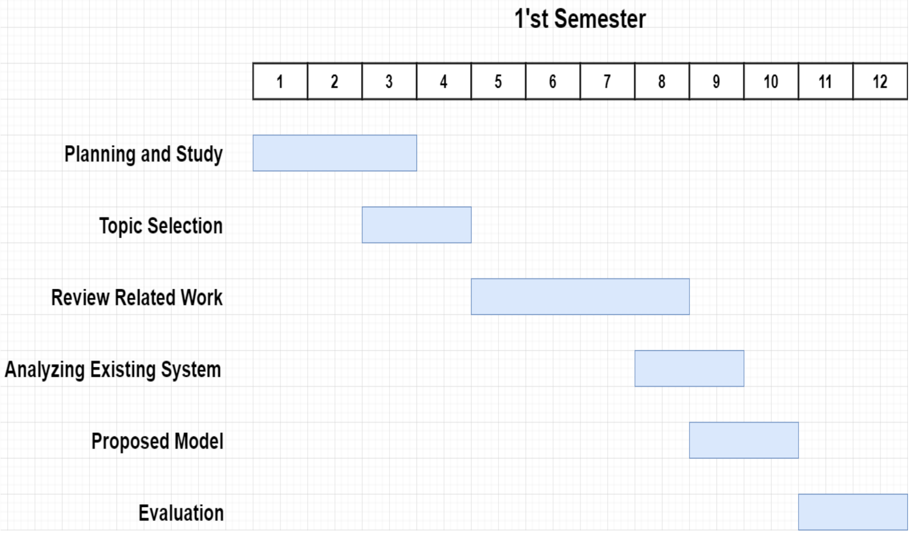

Figure 5.8: Gantt Chart I

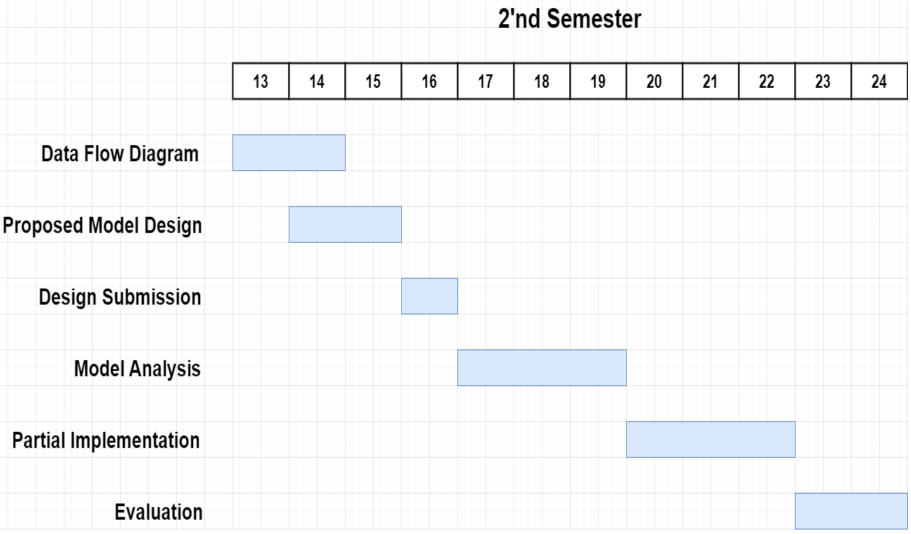

Figure 5.9: Gantt Chart II

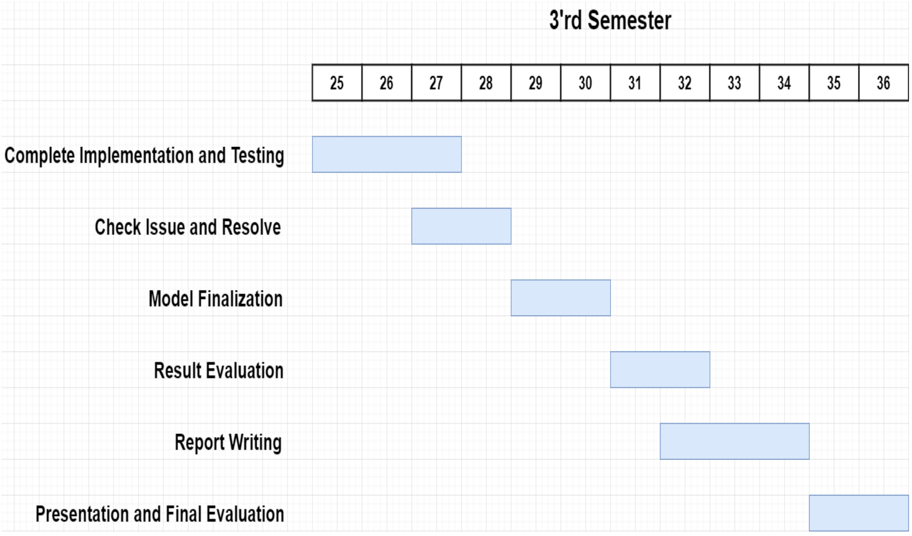

Figure 5.10: Gantt Chart III

5.8 Summary

However, this chapter provides a concise overview of the standards, impacts, ethical considerations, and challenges associated with the thesis work. Additionally, it presents the limitations, alternatives, schedules, tasks, and milestones integral to the proposed research.

Chapter 6
Conclusion and Future Work

6.1 Introduction

In this thesis paper, we proposed a novel and robust machine-learning based Phishing URL Detection system. We posit that our endeavors establish a formidable first layer of defense against deceptive URLs, as elucidated through the featured devices. Our illustration underscores the impactful capabilities of a straightforward machine-learning model in effectively identifying deceptive URLs. Moving forward, our focus is directed towards assessing methods to fortify our system against novel deceptive URL techniques, enhancing its overall resilience through innovative training approaches.

6.2 Future Works and Limitations

This section outlines the potential avenues for advancing the research on Phishing URL Detection using Machine Learning while acknowledging certain constraints. Moving forward, the integration of more sophisticated machine learning algorithms and the exploration of deep learning techniques hold promise for further improving detection accuracy and adaptability to emerging phishing tactics. Additionally, incorporating real-time learning mechanisms can enhance the system's ability to dynamically evolve with the evolving threat landscape.

However, limitations exist, such as the reliance on static datasets that may not fully capture the dynamic nature of phishing attacks. Addressing this constraint involves continuous data collection and augmentation strategies to ensure the model's robustness. Furthermore, the current research primarily focuses on URL-based detection, and future work could explore multi-modal approaches that integrate additional features like webpage content analysis for a more comprehensive threat assessment.

In terms of computational constraints, efforts to optimize the model for efficiency and scalability are essential. Collaboration with cybersecurity experts and industry stakeholders can provide valuable insights and real-world data, enriching the research's practical applicability. Despite these challenges, the future trajectory involves refining the model's sophistication, expanding its capabilities, and fostering collaboration for a holistic and effective defense against phishing threats.

6.3 Conclusion

Phishing is one of the most dangerous cybercrime today. Many cybercriminals use phishing to collect the sensitive and personal information of internet users. In this research paper, we represented new state-of-the-art phishing detection approaches using a machine learning model. In our research, we found the highest 86.9% accuracy using the XGBoost machine learning model. Other models like Decision Tree, Logistic Regression, Random Forest, and K-Nearest Neighbors also give good results in classifying phishing URLs.we also find that the Logistic Regression model performance increases if we increase the training dataset and the Decision Tree model gives a better result when the dataset is small. In the future, we want to increase our model accuracy and build a website that can take URLs from users and detect whether the given URL is phishing or not.

References

[1] Steve Sheng, Brad Wardman, Gary Warner, Lorrie Cranor, Jason Hong, and Chengshan Zhang. An empirical analysis of phishing blacklists. 2009.

[2] Justin Ma, Lawrence K Saul, Stefan Savage, and Geoffrey M Voelker. 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, pages 1245–1254, 2009.

[3] Justin Ma, Lawrence K Saul, Stefan Savage, and Geoffrey M Voelker. Identifying suspicious urls: an application of large-scale online learning. In Proceedings of the 26th annual international conference on machine learning, pages 681–688, 2009.

[4] RSA. Rsa fraud report. https://www.infopoint-security.de/ media/RSA_Fraud _Report_Q2_2020.pdf, 2020. [Online; accessed 08- January-2022].

[5] Broadcom. bistr main report v19. http://www.symantec.com/ content/en/us/ enterprise/other_resources/bistr_main_ report_v19_21291018.en-us.pdf., 2015. [Online; accessed 08- January-2022].

[6] RSA. Anti-fraud command center. rsa monthly online fraud report. http://www.emc.com/collateral/fraud-report/ rsa-online-fraud-report-012014.pdf, 2014. [Online; accessed 08-January-2022].

[7] Aaron Blum, Brad Wardman, Thamar Solorio, and Gary Warner. Lexical feature-based phishing URL detection using online learning. In Proceedings of the 3rd ACM Workshop on Artificial Intelligence and Security, pages 54–60, 2010.

[8] Ian Fette, Norman Sadeh, and Anthony Tomasic. Learning to detect phishing emails. In Proceedings of the 16th International Conference on World Wide Web, pages 649–656, 2007.

[9] Daisuke Miyamoto, Hiroaki Hazeyama, and Youki Kadobayashi. An evaluation of machine learning-based methods for detection of phishing sites. In International Conference on Neural Information Processing, pages 539–546. Springer, 2008.

[10] Colin Whittaker, Brian Ryner, and Marria Nazif. Large-scale automatic classification of phishing pages. 2010.

[11] Yue Zhang, Jason I Hong, and Lorrie F Cranor. Cantina: a content- based approach to detecting phishing web sites. In Proceedings of the 16th international conference on World Wide Web, pages 639–648, 2007.

[12] Madhusudhanan Chandrasekaran, Krishnan Narayanan, and Shambhu Upadhyaya. Phishing email detection based on structural properties. In NYS cyber security conference, volume 3. Albany, New York, 2006.

[13] Brad Wardman and Gary Warner. Automating phishing website iden- tification through deep md5 matching. In 2008 eCrime Researchers Summit, pages 1–7. IEEE, 2008.

[14] Christian Ludl, Sean McAllister, Engin Kirda, and Christopher Kruegel. On the effectiveness of techniques to detect phishing sites. In In- ternational Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, pages 20–39. Springer, 2007.

[15] Eric Medvet, Engin Kirda, and Christopher Kruegel. Visual-similarity- based phishing detection. In Proceedings of the 4th international con- ference on Security and privacy in communication netowrks, pages 1–6, 2008.

[16] Pawan Prakash, Manish Kumar, Ramana Rao Kompella, and Minaxi Gupta. Phishnet: predictive blacklisting to detect phishing attacks. In 2010 Proceedings IEEE INFOCOM, pages 1–5. IEEE, 2010.

[17] Mark Felegyhazi, Christian Kreibich, and Vern Paxson. On the potential of proactive domain blacklisting. LEET, 10:6–6, 2010.

[18] kaggle. Website phishing dataset. https://www.kaggle.com/ ahmednour/website-phishing-data-set., 2020. [Online; accessed 05-January-2022].

[19] Tin Kam Ho. Random decision forests. In Proceedings of 3rd inter- national conference on document analysis and recognition, volume 1, pages 278–282. IEEE, 1995.

[20] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning. springer series in statistics. In :. Springer, 2001. 23

[21] Guido Van Rossum et al. Python, 1991.

[22] Stefan Van Der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. Computing in science & engineering, 13(2):22–30, 2011.