INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 10, Issue 1 - V10I1-1211)
Available online at: https://www.ijariit.com

Prediction and detection of malicious URL using machine learning

Ibukunoluwa D. Okunnuga
maverickdeborah05@gmail.com
Austin Peay State University, TN 37044, United States

ABSTRACT

The efficient identification of malicious URLs has become crucial due to their growing hazard to individuals, companies, and digital infrastructure. This study evaluated multiple machine learning algorithms for their ability to predict and identify dangerous URLs. The research focused on the Random Forest Classifier since it outperformed rival models in binary and multi-class classification tasks. With 98.9% accuracy in binary classification, the Random Forest Classifier performed well. This shows the classifier can identify safe and hazardous URLs. The system's precision of 98.8%, F1 score of 99.3%, true positive rate of 99.7%, and true negative rate of 95.6 demonstrate its dependability. Multi-class classification accuracy was 97.0%, precision, recall, and F1 scores were good again for the Random Forest Classifier. This research provides practical tips for enhancing web security and shows how transparent AI models and interdisciplinary teamwork may solve complicated cybersecurity problems. This research has made a significant contribution to the body of known information, and its significance lies in the fact that it provides both benefits.

Keywords: *Malware, Machine learning, URL, Malicious.*

1. INTRODUCTION

In today's fast-paced digital landscape, the presence of Uniform Resource Locators (URLs) leading to malicious websites represents a substantial and ever evolving cybersecurity threat. Cybercriminals exhibit remarkable ingenuity in crafting these deceptive URLs, skillfully camouflaging them to resemble reliable and reputable sources. Regrettably, these seemingly trustworthy URLs serve as bait to lure unsuspecting users into their perilous trap, with the nefarious intent of pilfering sensitive personal information (Alomari et al., 2023). The repercussions of interacting with such malicious URLs can be dire and far-reaching. Once an individual inadvertently ventures onto these sinister sites, a multitude of harms can befall them. Personal and financial data, considered sacrosanct in today's digital age, is at grave risk of being plundered. Moreover, these rogue URLs can act as vectors for the dissemination of insidious malware, capable of infiltrating not only individual computers but entire networks, thereby compounding the damage exponentially. One of the most vexing challenges in combating this menace is the growing proliferation of rogue websites and the escalating complexity of cyberattacks (Naim et al., 2023). The deceptive artistry employed by cybercriminals makes it increasingly arduous for ordinary users to distinguish between safe and perilous web addresses, further intensifying the jeopardy. In light of this grim reality, safeguarding online safety and security has assumed paramount significance. It is imperative that robust and stringent

cybersecurity measures are implemented, ranging from fortified firewalls to cutting-edge threat detection technology. Equally crucial is the imperative for user education, as an informed user base is better equipped to identify and avoid the snares laid out by malicious URLs. The need for a collective effort, involving individuals, organizations, and governments, is clear and pressing. In summation, the widespread proliferation of harmful URLs has ushered in an era where digital denizens must remain vigilant and proactive. The constant evolution of cyber threats necessitates a dynamic and responsive approach to cybersecurity. As such, defenses must be fortified, cultivate a culture of cyber literacy, and harness state-of-the-art technology to mitigate the perils posed by these virtual traps (Martinez, 2019). The preservation of our digital well-being and the security of our personal data depend on it.

In light of the shortcomings of traditional reactive tactics, this innovative study is a giant step forward in the fight against rogue URLs, machine learning, and more specifically supervised learning techniques, present a proactive and flexible method to detecting and predicting these risks in real time. This change is consistent with the ever-changing character of cyber threats, since malicious URLs are always adapting to avoid conventional detection techniques (Prabakaran et al., 2023). Machine learning is a powerful tool against the everevolving strategies of hackers because of its capacity to rapidly scan large datasets, discover subtle trends, and adjust in real time. Taking this preventative measure helps businesses and individuals stay one step ahead of cybercriminals, reduces the likelihood of damaging data breaches, and strengthens security across the board in the digital world.

The primary purpose of this research is to advance the field of cybersecurity by leveraging machine learning techniques, specifically supervised learning approaches, to proactively detect and predict malicious URLs. By employing a diverse set of machine learning models and comprehensive evaluation metrics, this study aims to enhance the accuracy and efficiency of malicious URL identification, thereby contributing to the ongoing evolution of cybersecurity practices and fortifying the digital realm against the ever evolving and pervasive threats posed by malicious web addresses (Patgiri et al., 2019).
The comprehensive suite of machine learning models employed in this study, from Random Forest and Logistic Regression to K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree (DT), reflects the versatility required to tackle this multifaceted problem (Smith & Brown, (2018). Furthermore, the choice of evaluation metrics, encompassing accuracy, precision, true positive rate (TPR), true negative rate (TNR), f1 score, Area under the Receiver Operating Characteristic Curve (ROC AUC), confusion matrix, and classification report, demonstrates a rigorous and holistic assessment of model performance. At the end of the analysis, the best model that detects and predicts accurately will be recommended as the model for the detection and prediction of malicious URLs (Sahoo, (2017).

By integrating machine learning into the arsenal against malicious URLs, this research contributes to the ongoing evolution of cybersecurity practices, fostering a more proactive and adaptable defense mechanism that can better cope with the ever-evolving landscape of online threats. This not only enhances the protection of individuals and organizations but also underscores the importance of collaborative efforts among cybersecurity experts and organizations in fortifying the digital realm against malicious intent (Adebowale et al., 2020).

PROBLEM STATEMENT

In today's interconnected digital landscape, the proliferation of malicious URLs has become a critical and evolving cybersecurity concern. These deceptive web addresses, engineered to mimic legitimate sites, pose substantial threats by enabling cybercriminals to infiltrate systems, steal sensitive data, and propagate malware. Traditional security measures, such as signature-based antivirus software and blacklisting known malicious domains, have proven to be increasingly ineffective as malicious URLs rapidly adapt and mutate to evade detection (Chen et al., 2019). This persistence of threats necessitates a proactive and adaptive approach to counter them. Thus, the central problem addressed by this research lies in the development and evaluation of machine learning-based models, including Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree (DT), to predict and detect malicious URLs accurately and efficiently in real-time, ultimately enhancing the cybersecurity infrastructure against this formidable threat (Brown et al., 2018).

AIM AND OBJECTIVES

The aim of this research is to develop machine learning models for the detection and prediction of malicious URLs. The specific objectives are to:

I. Build and train machine learning models, including Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree (DT), to achieve real-time detection and prediction of malicious URLs.

ii. Assess model performance comprehensively by employing a diverse set of classification metrics, such as accuracy, precision, true positive rate (TPR), true negative rate (TNR), f1 score, Area under the Receiver Operating Characteristic Curve (ROC AUC), confusion matrix, and classification report.

iii. Enhance the cybersecurity infrastructure by integrating the developed models into the defense mechanisms against the dynamic and evolving threat landscape posed by malicious URLs.

iv. Contribute to the broader field of cybersecurity by providing a nuanced understanding of the strengths and limitations of machine learning-based approaches for combating malicious URLs, thereby aiding in the fortification of digital systems and networks.

RESEARCH QUESTIONS

The following research questions will be used to achieve the aim and objectives of this research.

I. How can machine learning models, including Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree (DT), be effectively developed, and trained for real-time malicious URL detection and prediction?

ii. What comprehensive set of classification metrics, such as accuracy, precision, true positive rate (TPR), true negative rate (TNR), f1 score, Area under the Receiver Operating Characteristic Curve (ROC AUC), confusion matrix, and classification report, should be employed to rigorously assess the performance of these machine learning models in the context of malicious URL detection?

iii. In what ways can the integration of these machine learning model into cybersecurity infrastructure enhance the defense mechanisms against the dynamic and evolving threat landscape posed by malicious URLs?

iv. What insights can be gained from the research to contribute to the broader field of cybersecurity, particularly in terms of understanding the strengths and limitations of machine learning-based approaches for combating malicious URLs and fortifying digital systems and networks?

2. RELATED WORKS

The landscape of malicious URL detection has been extensively explored in previous research and projects, reflecting the pressing need to combat the ever evolving threats posed by deceptive web addresses. A multitude of studies and projects have delved into the development and evaluation of innovative approaches to identify and mitigate the risks associated with malicious URLs. A noteworthy body of research has been dedicated to machine learning-based techniques for predictive and proactive threat detection, showcasing the efficacy of these methods in bolstering cybersecurity.

According to Liu et al. (2019), numerous studies have demonstrated the effectiveness of supervised machine learning models in malicious URL detection.

These models have been extensively trained on large datasets containing both malicious and benign URLs, enabling them to discern intricate patterns and characteristics that distinguish between the two categories. Feature extraction techniques, such as the analysis of URL structure, content, and behavioral attributes, have significantly enhanced the performance of these models. The research has showcased the potential of various supervised learning algorithms, including Random Forest, Support Vector Machines (SVM), and deep neural networks, in accurately classifying malicious URLs and bolstering cybersecurity defenses.

Additionally, as highlighted by Sharma et al. (2018), the integration of real-time threat intelligence and dynamic analysis of URLs has become a central focus in the pursuit of effective malicious URL detection. This approach involves leveraging

threat feeds, URL reputation services, and sandboxing techniques to continuously update and validate the threat status of URLs. By harnessing real-time information, these methodologies aim to outmaneuver cybercriminals who frequently alter their tactics to evade detection. The synthesis of these research efforts emphasizes the critical importance of a multifaceted approach to malicious URL detection, combining the strengths of machine learning, threat intelligence, and dynamic analysis to fortify cybersecurity defenses.

Moreover, Bhuyan et al. (2020) have emphasized the increasing importance of explainable AI (XAI) and interpretable machine learning models in the field of malicious URL detection. These researchers recognize the significance of transparency and comprehensibility in cybersecurity decision-making.

Consequently, their studies delve into the development of models that provide insights into their decision processes. This empowers cybersecurity experts to understand why a particular URL is classified as malicious or benign. This focus on explainability not only enhances trust in the models but also assists in refining detection strategies and improving the overall effectiveness of malicious URL detection systems.

According to Anderson (2020), supervised machine learning has played a pivotal role in tackling the complex challenge of malicious URL detection. These models are trained on diverse datasets, encompassing a wide range of malicious and benign URLs. The training process equips the algorithms with the ability to discern subtle patterns and characteristics unique to malicious URLs. Features such as URL structure, content analysis, and lexical properties have been extensively explored to enhance the discriminative power of these models. This research demonstrates that supervised learning algorithms, including Random Forest, SVMs, and deep neural networks, offer a robust foundation for classifying and identifying malicious URLs in real-time, bolstering the cybersecurity landscape.

Furthermore, research by Ramanauskaite and Garsva (2019) highlights the significance of real-time threat intelligence integration in the battle against malicious URLs. Their studies emphasize the dynamic nature of cyber threats, with adversaries continually altering their strategies to evade detection. To address this, cybersecurity practitioners have turned to threat feeds, URL reputation services, and dynamic sandboxing techniques. These mechanisms provide up-tothe-minute insights into the trustworthiness of URLs, enabling rapid response to emerging threats. The synergy of real-time threat intelligence and machine learning-based detection fosters a proactive defense strategy capable of staying ahead of cybercriminals and safeguarding digital assets.

Moreover, as stated by Feng et al. (2020), the pursuit of explainable AI (XAI) and interpretable machine learning models has gained prominence in the context of malicious URL detection. Their research underlines the importance of transparency and interpretability in cybersecurity decision-making. XAI models provide insights into the rationale behind their predictions, enabling cybersecurity experts to understand the factors contributing to a URL's classification as malicious or benign. This fosters not only greater trust in the models but also the ability to fine-tune detection strategies and improve overall system performance, ensuring more robust protection against deceptive web addresses.

In summary, the review of previous research in malicious URL detection uncovered a wealth of insights and innovative methodologies. Supervised machine learning models have consistently proven their effectiveness, demonstrating the capability to accurately differentiate malicious and benign URLs by capturing intricate patterns and features. These models, encompassing algorithms like Random Forest, Support Vector Machines (SVM), and deep neural networks, have played a pivotal role in bolstering cybersecurity defenses. Additionally, the integration of real-time threat intelligence, including threat feeds, URL reputation services, and dynamic sandboxing, has emerged as a proactive strategy to counter the dynamic tactics of cybercriminals. This approach provides timely updates on URL trustworthiness, allowing rapid responses to emerging threats. Furthermore, the emphasis on explainable AI (XAI) and interpretable models has improved the transparency of decision processes, empowering cybersecurity experts to refine detection strategies and enhance overall system performance. Together, these findings and methodologies contribute to a more adaptive and robust defense against the pervasive threat of malicious URLs. The table below gives a clear view of the related work as discussed above.

3. METHODOLOGY

The methodology section provides a detailed plan for this research, which uses machine learning to predict and detect bad URLs, a crucial cybersecurity task. This chapter carefully describes the research's structured strategy, including data collecting, preprocessing, model creation, evaluation, and ethical concerns, ensuring its accuracy and trustworthiness in addressing cybersecurity issues. The study compares machine learning algorithms for malicious URL identification using a comparative research approach and exploratory analysis. Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree are used in supervised learning. Implementation uses Python and tools like pandas, scikit-learn, and Keras. The study uses the ISCX-URL2016 dataset and PCA for feature selection to improve model efficiency. To evaluate the models, accuracy, precision, recall, F1-score, ROC AUC, confusion matrices, and classification reports were used.

Research Design

This study uses a comparative approach to evaluate and compare machine learning models for detecting malicious URLs in cybersecurity. It uses a range of algorithms, including Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree, each chosen for their unique characteristics and capabilities. The research is grounded in prior research and supervised learning contexts, ensuring a rigorous exploration of their performance.

Research Method

This study uses Python programming to analyze and implement machine learning algorithms for detecting malicious URLs. It uses libraries like pandas, scikit-learn, and Keras to bridge theory and practice. The methodical approach ensures the research's efficacy in detecting malicious URLs, allowing for rigorous optimization and refinement of models. This approach ensures the research's practical application in real-world applications.

Data Collection Technique

The data collection technique employed in this research centers around the utilization of the ISCX-URL2016 dataset, which can be sourced from the official repository at "https://www.unb.ca/cic/datasets/url-2016.html." This dataset constitutes a rich and extensive resource, meticulously curated to encompass a diverse collection of URLs that have been thoughtfully labeled as either malicious or benign. Serving as the bedrock upon which the entire research is built, this dataset provides the fundamental raw material necessary for both training and evaluating the machine learning models. Its comprehensive nature and welldefined labeling ensure the research's robustness and reliability, enabling a thorough exploration of the models' performance in the crucial task of malicious URL detection.

Data Preprocessing

Data preprocessing is a fundamental step that occurs before the actual model training begins. In this phase, the dataset, sourced from the ISCX-URL2016 repository, undergoes a systematic and thorough transformation process. Primarily, missing values are addressed meticulously to ensure that the dataset is complete and devoid of gaps. Data cleaning procedures are applied rigorously to rectify any anomalies, errors, or inconsistencies within the dataset, guaranteeing its integrity and reliability. Moreover, as machine learning models typically require numerical input, categorical variables present in the dataset are encoded into a suitable numeric format. This transformation ensures that the dataset is in an optimal state for machine learning analysis, facilitating accurate model training and robust evaluation. Overall, data preprocessing plays a critical role in refining the dataset, setting the stage for successful and meaningful outcomes in the research's machine learning endeavors (Matsuzaka & Uesawa (2023).

Feature Selection and Engineering

This research optimizes the machine learning process by reducing dimensionality using Principal Component Analysis (PCA). PCA identifies significant dimensions in datasets, streamlining computational burden and enhancing model efficiency. It is suitable for large-scale datasets like ISCX-URL2016. The research also employs feature engineering techniques to extract valuable insights from existing ones. These techniques work together to optimize dataset quality and comprehensiveness, contributing to a robust machine learning framework for malicious URL detection.

Analysis and Implementation

The analysis and implementation phase marks a pivotal juncture in the research journey, where the selected machine learning models come to life through practical implementation. In this dynamic phase, the research leverages Python's versatile scikit-learn and Keras libraries to breathe functionality into the chosen models: Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree (DT) (Sahu et al., 2023). Each of these models is meticulously executed, a process that entails the training of the models on the preprocessed dataset, followed by rigorous validation procedures to ensure their effectiveness and generalization. The phase also involves a fine-tuning process where model parameters are systematically adjusted and optimized to attain the pinnacle of performance. The end goal, as articulated by Martinez (2019), is to harness the full potential of these machine learning models, ensuring that they operate at their peak in the critical task of malicious URL detection. This phase thus transforms the theoretical constructs into practical, functional tools, laying the groundwork for a comprehensive evaluation of their performance and efficacy.

Result Evaluation and Discussion

Following the training and rigorous evaluation of the machine learning models, the research embarked on a crucial phase of result analysis and discussion. The primary objective was to assess the predictive capabilities of various machine learning algorithms in the context of malicious URL detection. To comprehensively evaluate model performance, an array of classification metrics was employed, including accuracy, precision, true positive rate (TPR), true negative rate (TNR), F1 score, Area under the Receiver Operating Characteristic Curve (ROC AUC), confusion matrix, and classification report. This thorough examination of each algorithm's performance provided insights into their effectiveness and reliability in recognizing malicious URLs accurately.

The results were subsequently contextualized within the operational landscapes of network security, shedding light on the practical implications and real-world applications of each algorithm, extending beyond mere quantitative metrics. These findings aim to empower stakeholders with the knowledge needed to make informed decisions for enhancing network security (Zhang and Wu, 2023).

Justification for Model Selection and Supervised Learning

This research's machine learning approach used supervised learning because it performs well in difficult categorization tasks. Models may learn from a labeled dataset and discriminate malicious and benign URLs using supervised learning, making it perfect for harmful URL detection. This meets research's prediction accuracy and real-world applicability aims. The five models—Random Forest, Logistic Regression, K-Nearest Neighbors, Linear Support Vector Machine, and Decision Tree—are purposely varied. Each malicious URL detection methodology has strengths and features for particular parts. While Logistic Regression is interpretable, Random Forest handles complex, high-dimensional data well. K-Nearest Neighbors uses instance-based learning, Linear Support Vector Machine handles linearly separable data well, and Decision Tree provides clear paths. This ensemble of models tests many modeling methodologies to better identify risky URLs.

4. DATA ANALYSIS AND IMPLEMENTATION

This session discusses the implementation of machine learning techniques for detecting malicious URLs. A comprehensive analysis of the dataset was conducted, focusing on data features, variable distributions, and data visualization methods. Two machine learning approaches were used: binary classification and multi-class classification. Various machine learning models were used, including Random Forest, Decision Tree, K-Nearest Neighbors, Logistic Regression, and Linear Support Vector Machine. A random seed value was established to ensure consistency. Principal Component Analysis (PCA) was applied to the dataset, and the effectiveness of the models was evaluated using various metrics. After a thorough evaluation, one model was identified as the most suitable for detecting malicious web addresses.

Data Cleaning

Data cleaning was done to ensure data accuracy and completeness for analysis and modeling. The dataset had 19,183 NaN or Infinity values in nine columns. Machine learning algorithms and statistical techniques cannot handle these values, so they were replaced with 0 to avoid errors and ensure smooth algorithm execution.

Exploratory Data Analysis

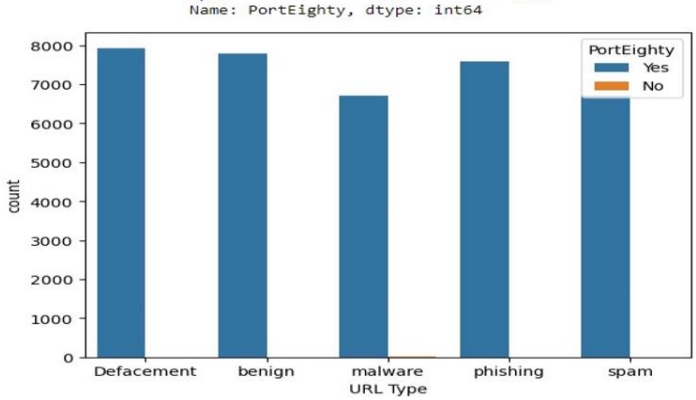
The dataset comprises 36,707 records and 80 attributes. These attributes can be classified as either numeric or categorical, as shown in Table 4.1 below. Numeric attributes represent quantitative data and can take on values that are either continuous (float) or discrete (integer). Categorical attributes represent qualitative data and comprise distinct categories. Table 4.1 also shows the number of unique values and the range of values for each attribute.

Table 4.1: Snippet of the Data type and Range of Features

S/N	Features	Data type	Unique	Range
1.	Querylength	Numeric	319	0...1385
2.	domain_token_count	Numeric	15	2...19
3.	path_token_count	Numeric	43	0...68
4.	avgdomaintokenlen	Numeric	162	1.5...29.5
5.	longdomaintokenlen	Numeric	52	2...63
6.	avgpathtokenlen	Numeric	835	0...105
7.	tld	Numeric	15	2...19
8.	charcompvowels	Numeric	150	0...193
9.	charcompace	Numeric	127	0...142
10.	ldl_url	Numeric	133	0...207
11.	ldl_domain	Numeric	20	0...37
12.	ldl_path	Numeric	132	0...207

The 'URL_Type_obf_Type' attribute serves as the target variable, and it classifies web addresses into five distinct categories. Among these categories, four pertain to different forms of malicious URLs, encompassing Defacement, Malware, Phishing, and Spam. The fifth category, known as Benign, encompasses regular and legitimate web addresses that do not carry any malicious intent.

```
URL_Type_obf_Type  PortEighty
Defacement         Yes          7929
                   No              1
benign             Yes          7781
malware            Yes          6701
                   No             11
phishing           Yes          7584
                   No              2
spam               Yes          6698
Name: PortEighty, dtype: int64
```



Malware, Phishing, and Spam. The fifth category, known as Benign, encompasses regular and legitimate web addresses that do not carry any malicious intent.

The figure below shows the number of URL types that use port 80. As evident from the graph, a significant majority of web addresses use port 80. Port 80 is commonly associated with HTTP (Hypertext Transfer Protocol), which is the foundation of data communication on the World Wide Web. Therefore, this figure below emphasizes the prevalence of HTTP-based web traffic on the Internet.

Figure 4.1: Distribution of URL Types using Port 80

Figure 4.2 below illustrates the average query length for different types of URLs.
The data is grouped by URL type, and each type's mean query length is displayed. The key observation from this figure is that Spam URLs have the highest average query length compared to other URL types.

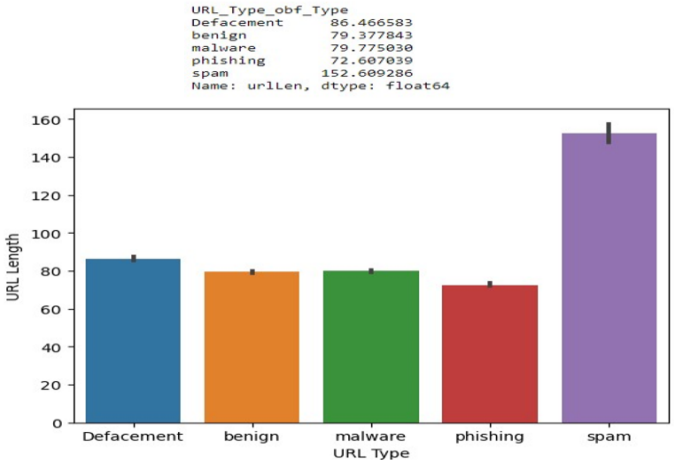
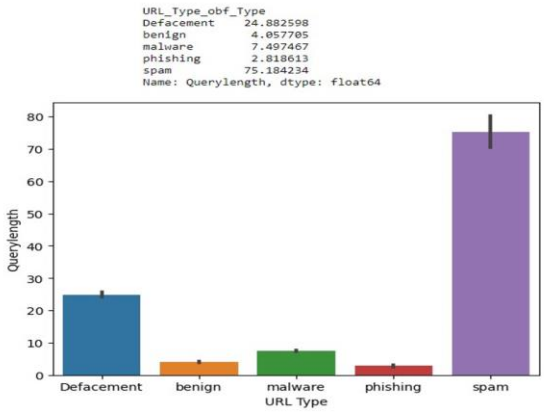


Figure 4.3 depicts the average URL length for different URL types. It is evident from the chart that the URL type labeled as 'Spam' exhibits the highest mean URL length among all the categories.

Figure 4.3: Mean URL Length per URL Type

Figure 4.4 is a box plot that conveys how path lengths are distributed across different URL types. The box plot reveals that, on average, Spam URLs tend to have longer path lengths when compared to the other URL types.



Thus, the data in Figures 4.2, 4.3, and 4.4 show that Spam URLs stand out due to their longer query, URL, and path lengths, suggesting a pattern of complexity and verbosity often associated with spammy websites.

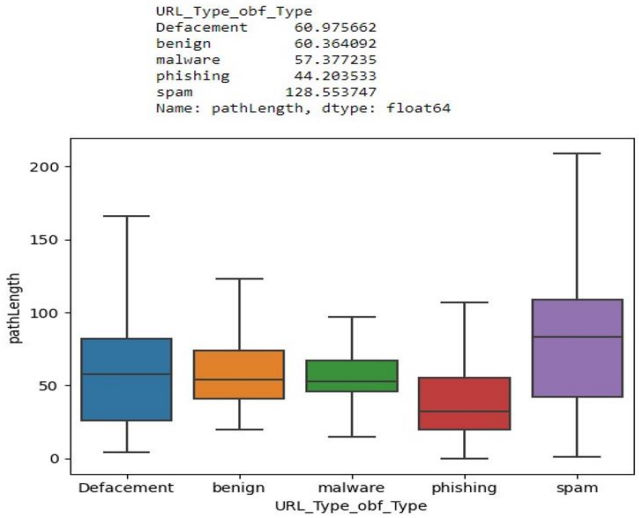


Figure 4.4: Distribution of Path Lengths by URL Type

Figure 4.5 illustrates a comparison of the maximum number of domain tokens within various types of URLs. Domain tokens are the individual parts or elements that make up web addresses.

Benign URLs, which have no malicious intent, typically consist of a minimal number of these components, with the highest count being just 3 tokens.

In contrast, Phishing URLs, which are used for malicious purposes like identity theft or fraud, exhibit a markedly complex structure. Some Phishing URLs are found to contain as many as 19 distinct elements or tokens. This complexity is a characteristic of Phishing URLs because they often try to mimic legitimate websites or hide their true intentions by using a convoluted structure.

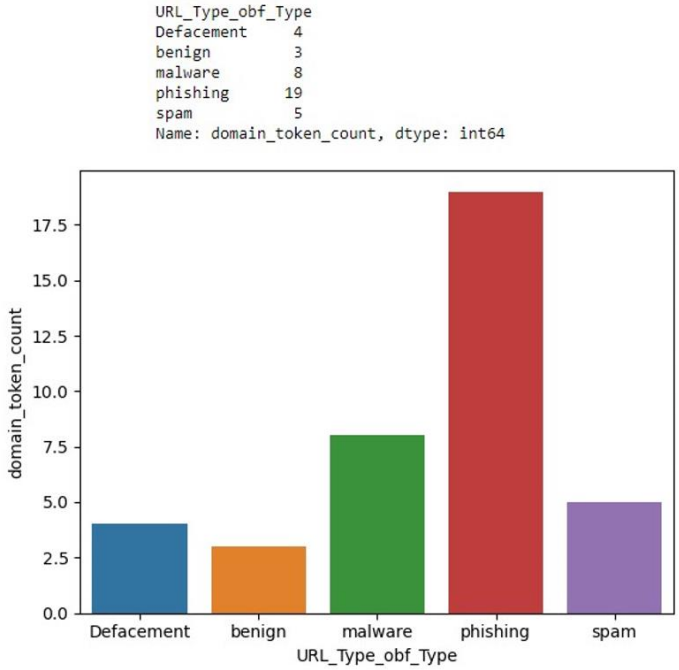


Figure 4.5: Maximum Domain Token Counts in URL Types

Feature Engineering

The dataset's target variable, a non-numeric attribute, was modified to represent numeric values using two new columns: 'Label_Binary' and 'Label_Multiple'. The 'Label_Binary' column allows binary classification of web addresses into safe and malicious categories, with 'Benign' labeling safe URLs and 'Malicious URLs' labeling malicious ones.


```
Malicious URLs    28926
Benign             7781
Name: Label_Binary, dtype: int64
```

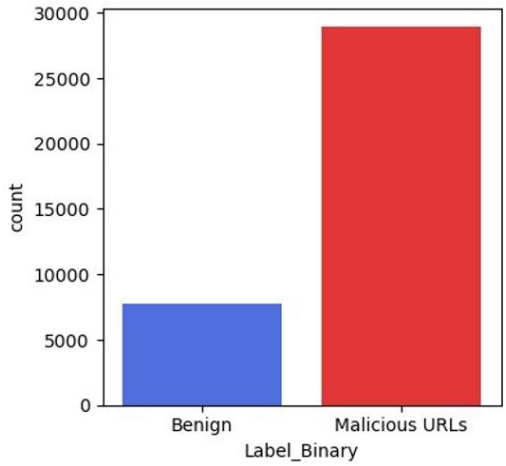


Figure 4.6: Binary Classification Labels

The multi-classification approach aimed to classify malicious web addresses into distinct categories using a new data column called 'Label_Multiple'. This column assigned numerical values (0, 1, 2, 3, 4) to distinguish between benign and malicious URLs, simplifying the training and assessment process for machine learning models.

```
Defacement    7930
benign        7781
phishing      7586
malware       6712
spam          6698
Name: URL_Type_obf_Type, dtype: int64
```

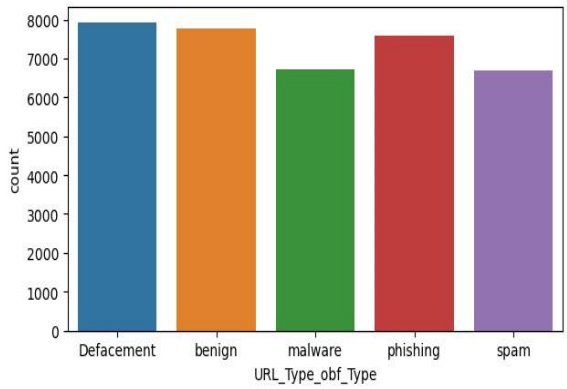


Figure 4.7: Multi-class Classification Labels

Dimensionality Reduction using Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a machine learning technique used to reduce dataset dimensionality while preserving key information. It transforms features into principal components, linear combinations of original ones, capturing the most variance in the data.

Binary Classification

After the models were trained using the training data, an evaluation was conducted using the test set, employing multiple metrics to measure their performance. The outcomes of these evaluations are presented in the table below.

Table 4.2: Evaluation Results of Binary Classification

Model	F1 Score	TPR	TNR	AUC ROC	Accuracy	Precision
Random Forest Classifier	0.989	0.988	0.993	0.997	0.956	0.977
Decision Tree Classifier	0.964	0.976	0.977	0.978	0.912	0.945
K-Neighbors Classifier	0.969	0.978	0.981	0.984	0.916	0.950
Logistic Regression	0.926	0.954	0.953	0.952	0.829	0.890
Linear SVC	0.906	0.942	0.940	0.939	0.785	0.862

The Random Forest Classifier outperformed other models in key aspects, achieving an impressive accuracy of 98.9%, high precision of 98.8%, and a remarkable F1 score of 99.3%. It also had a high true positive rate of 99.7% and a true negative rate of 95.6%, indicating its reliability in correctly classifying negative cases. The K-Neighbors Classifier followed closely behind, achieving a strong accuracy of 96.9%, precision of 97.8%, F1 score of 98.1%, and AUC ROC score of 0.950. The Decision Tree Classifier demonstrated remarkable performance with an accuracy of 96.4%, precision of 97.6%, and F1 score of 97.7%. The Linear SVC model had the lowest accuracy of 90.6%, but demonstrated commendable precision of 94.2% and F1 score of 94.0%.

Figure 4.8 illustrates the accuracy scores of the several machine learning models. In terms of accuracy ranking, the Random Forest Classifier has the highest accuracy score, making it the top-performing model. The K-Nearest Neighbors Classifier and the Decision Tree Classifier follow closely in the second and third positions, respectively. Logistic Regression is rated fourth, and the Linear Support Vector Classifier is ranked last in terms of accuracy.

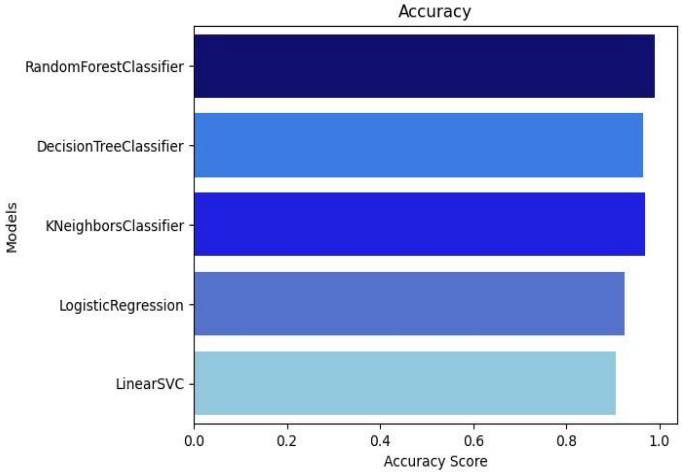
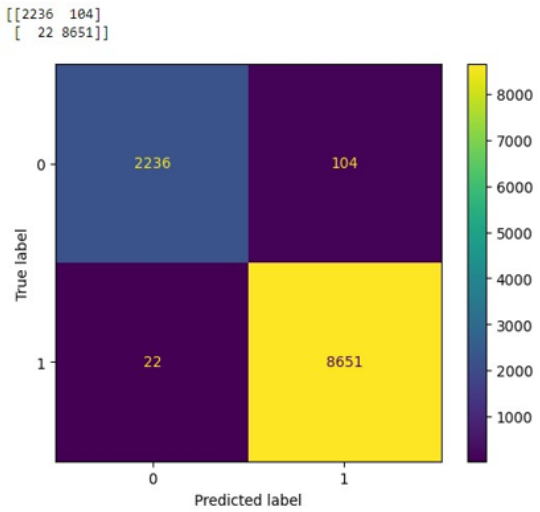


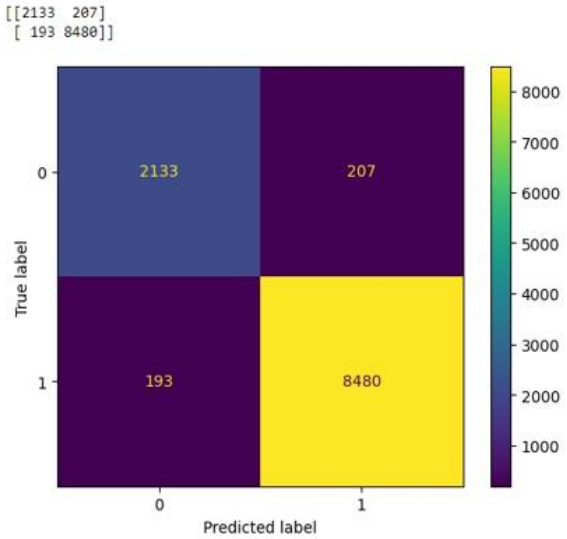
Figure 4.8: Accuracy Scores for Binary Classification Models

Confusion Matrices

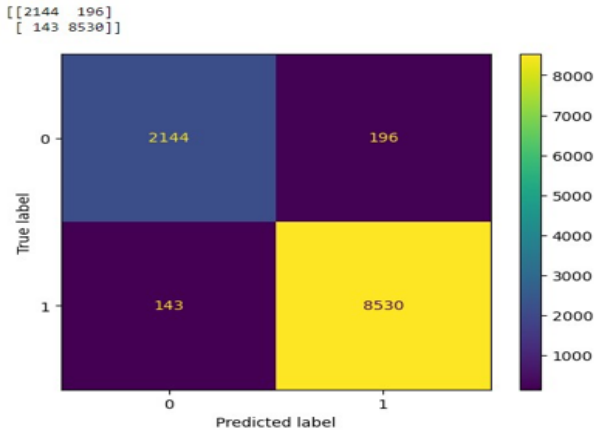
Figure 4.9 a-d presents confusion matrices that organize the predictions made by various models into four distinct categories: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). These matrices visualize the results of different classifiers when distinguishing between malicious URLs and safe URLs. The performance of each classifier is evaluated by considering how many instances are inaccurately labeled as either malicious web addresses or safe web addresses.



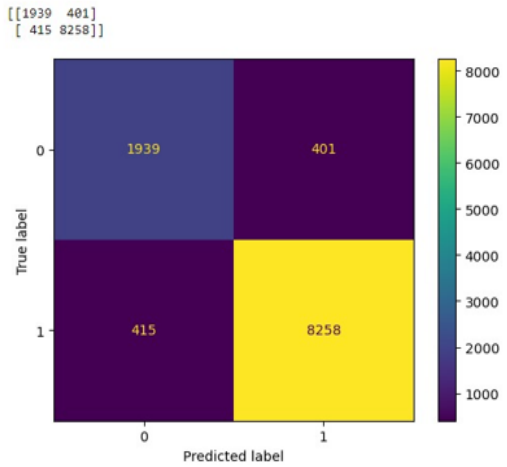
a) Random Forest



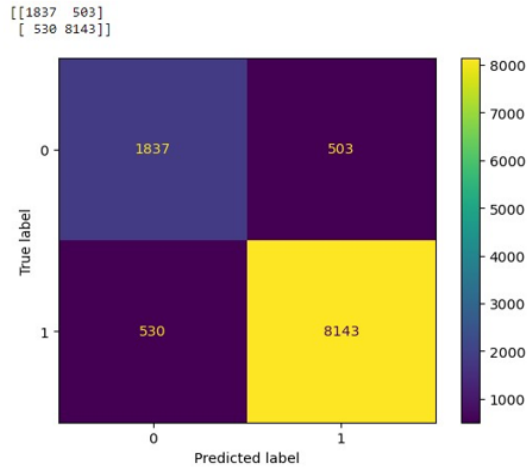
b) Decision Tree



c) K-Neighbors Classifier



d) Logistic Regression



e) Linear SVC

Figure 4.9: Confusion Matrices for Binary Classification Models

The Random Forest Classifier incorrectly labeled 104 legitimate web addresses as malicious, and 22 actual malicious web addresses as normal.

The Decision tree Classifier made an error by misclassifying 207 safe web addresses as Malicious URLs, while also erroneously categorizing 193 Malicious URLs as legitimate web addresses.
In the same vein, the K-Neighbors Classifier made erroneous predictions by mistakenly categorizing 196 web addresses as safe when they were malicious URLs. Additionally, it also incorrectly classified 143 instances of malicious URLs as safe web addresses.

The Logistic Regression model's performance was suboptimal as it made a
significant number of errors in its predictions. Specifically, it incorrectly classified 401 legitimate web addresses as malicious URLs, and in 415 instances, it wrongly categorized malicious URLs as safe web addresses.
The Linear Support Vector Classifier exhibited the highest number of incorrect predictions. Specifically, it erroneously identified 503 legitimate web addresses as malicious URLs and also incorrectly categorized 530 instances of malicious URLs as safe web addresses.

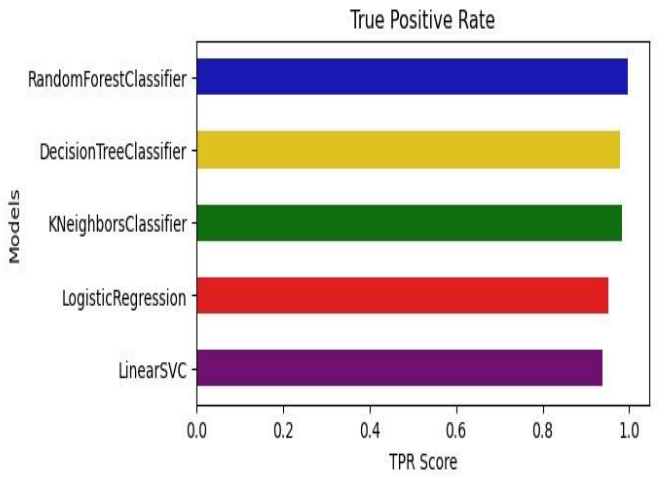


Figure 4.10 displays the True Positive Rate for the binary classification models. This metric measures the model's ability to accurately identify instances of Malicious

URLs among all the actual instances of Malicious URLs. The Random Forest Classifier achieved the highest TPR, performing exceptionally well with a rate of
99.7%, followed by the K-Neighbors Classifier (98.4%), the Decision Tree Classifier (97.8%), the Logistic Regression model (95.2%), and lastly the Linear Support Vector Classifier (93.9%).
Figure 4.10: True Positive Rate (TPR) for Binary Classification Models.

TNR and TPR Binary Classification

Figure 4.11 shows the TNR values for different binary classification models which assesses how well these models identify safe URLs (those that are not Malicious URLs) correctly, as a percentage of all the actual safe URLs. The Random Forest Classifier achieved the highest TNR of 95.6%, followed by the K-Neighbors Classifier (91.6%), the Decision Tree Classifier (91.2%), the Logistic Regression model (82.9%), and finally the Linear SVC (78.5%).

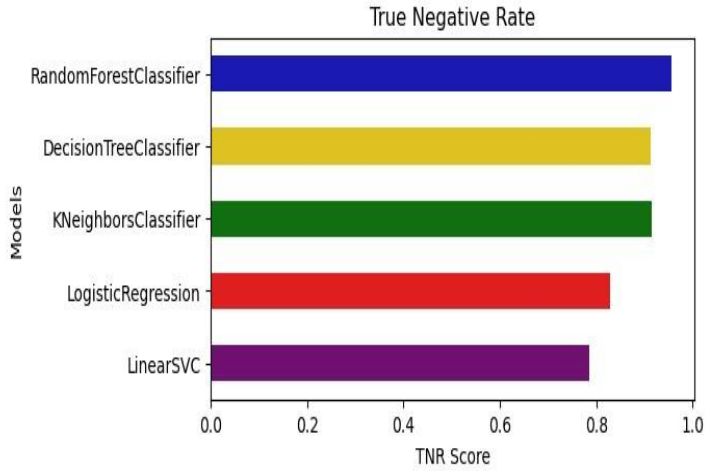


Figure 4.11: True Negative Rate (TNR) for Binary Classification Models.

AUC ROC Score for Binary Classification

In Figure 4.12, there is a plot illustrating the Area Under the Receiver Operating Characteristic Curve (AUC ROC). This curve serves as a tool for evaluating the performance of different models when it comes to classifying data into two distinct categories. Among these models, the Random Forest Classifier stands out with the highest AUC ROC score, which is recorded at 0.98.

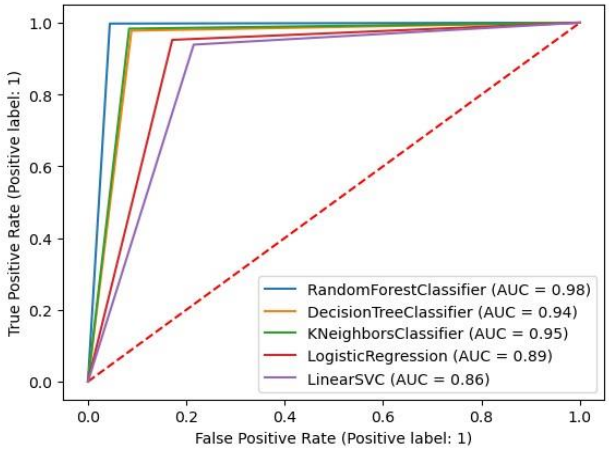
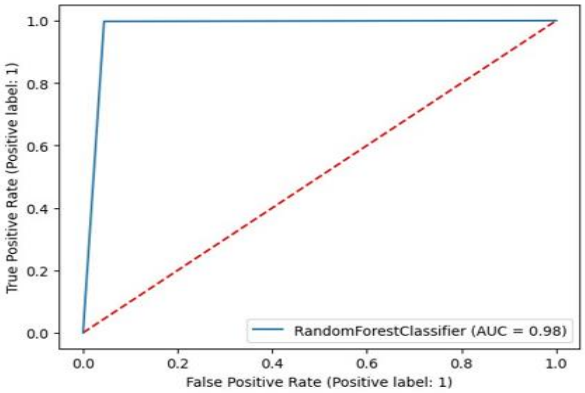
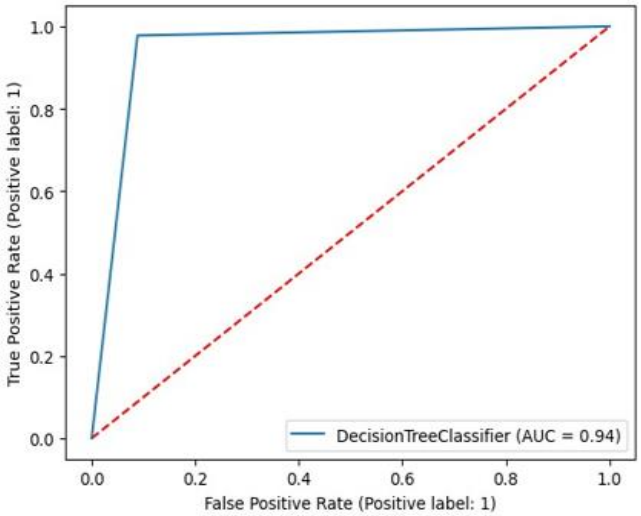


Figure 4.12: Joint ROC Curve for Binary Classification Models

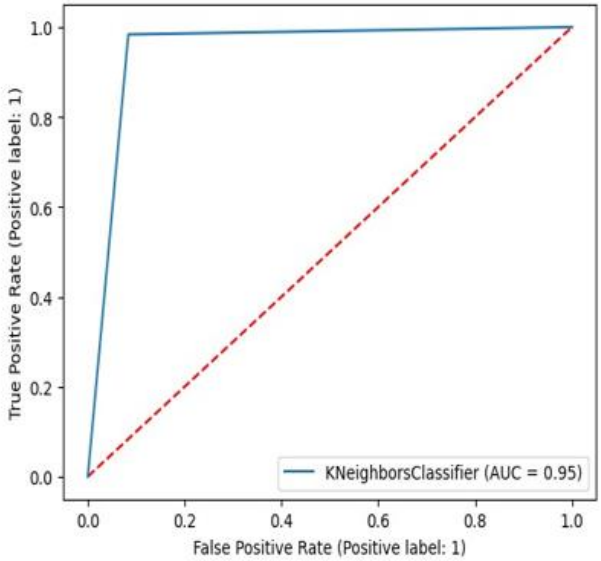
Figure 4.13 a-d illustrates the individual AUC ROC curve of each model.



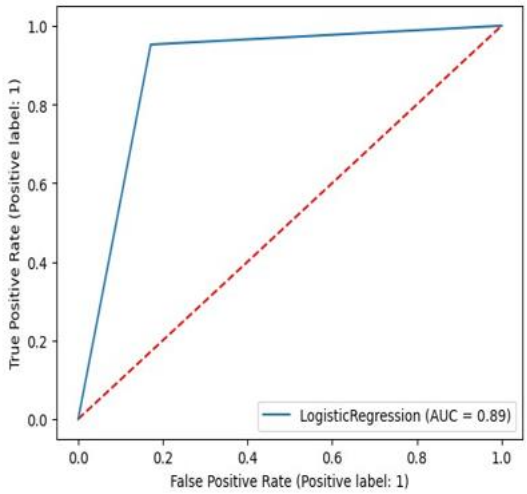
a) Random Forest Classifier



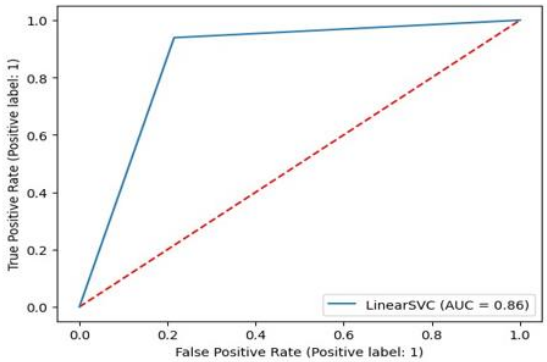
b) Decision Tree Classifier



c) K-Neighbors Classifier



d) Logistic Regression



e) Linear SVC

Figure 4.13: Individual ROC Curves for Binary Classification Models

Multi-class Classification

The table below shows the accuracy scores achieved by the different models that were employed for this multi-class classification task, enabling the comparison of the performance of various classifiers.
Table 4.3: Accuracy Scores of Multi-class Classification

Model	Accuracy
Random Forest Classifier	0.970
Decision Tree Classifier	0.916
K-Neighbors Classifier	0.916
Logistic Regression	0.807
Linear SVC	0.721

As shown in the figure below, the Random Forest Classifier achieved the highest accuracy score of 0.970, indicating that it successfully classified data into multiple classes with a very high level of accuracy. In contrast, the Linear SVC was the least accurate among the models with an accuracy score of 0.721.

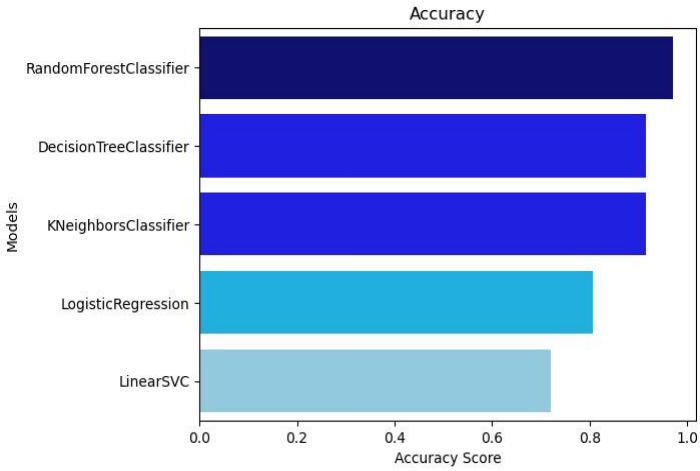
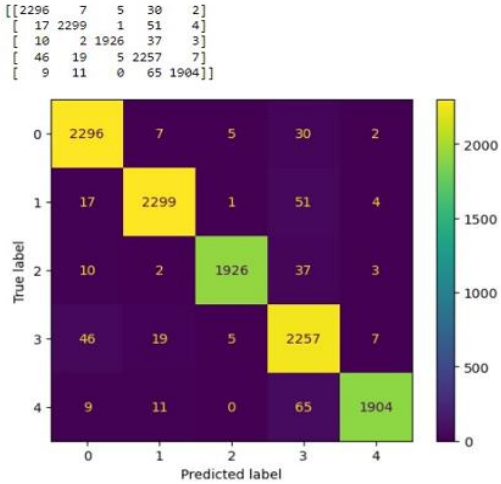


Figure 4.14: Accuracy Scores for Multi-class Classification Models

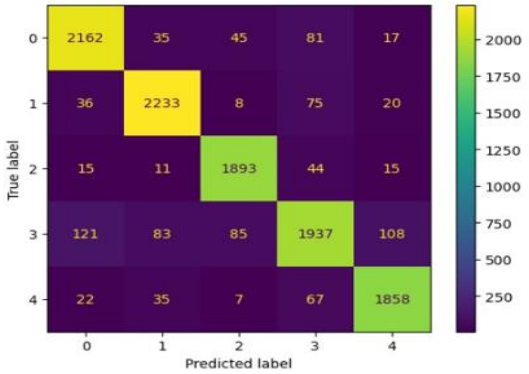
Confusion Matrices for Multi-Class Classification

Figure 4.15 a-d showcases five confusion matrices which represent the outcomes of predictions made by different models for each class.



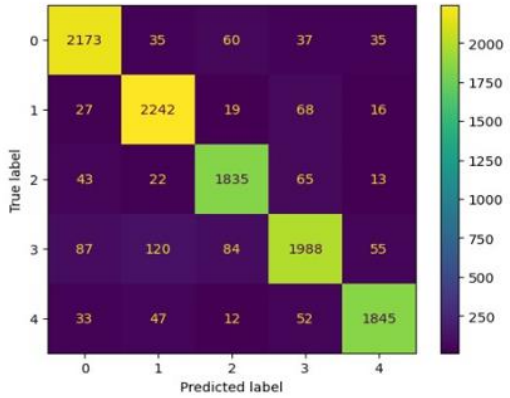
a) Random Forest Classifier

```
[[2162   35   45   81   17]
 [  36 2233    8   75   20]
 [  15   11 1893   44   15]
 [ 121   83   85 1937  108]
 [  22   35    7   67 1858]]
```



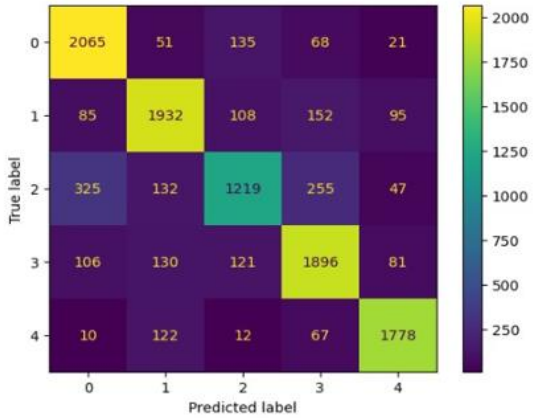
b) Decision Tree Classifier

```
[[2173   35   60   37   35]
 [  27 2242   19   68   16]
 [  43   22 1835   65   13]
 [  87  120   84 1988   55]
 [  33   47   12   52 1845]]
```



c) K-Neighbors Classifier

```
[[2065   51  135   68   21]
 [  85 1932  108  152   95]
 [ 325  132 1219  255   47]
 [ 106  130  121 1896   81]
 [  10  122   12   67 1778]]
```



d) Logistic Regression

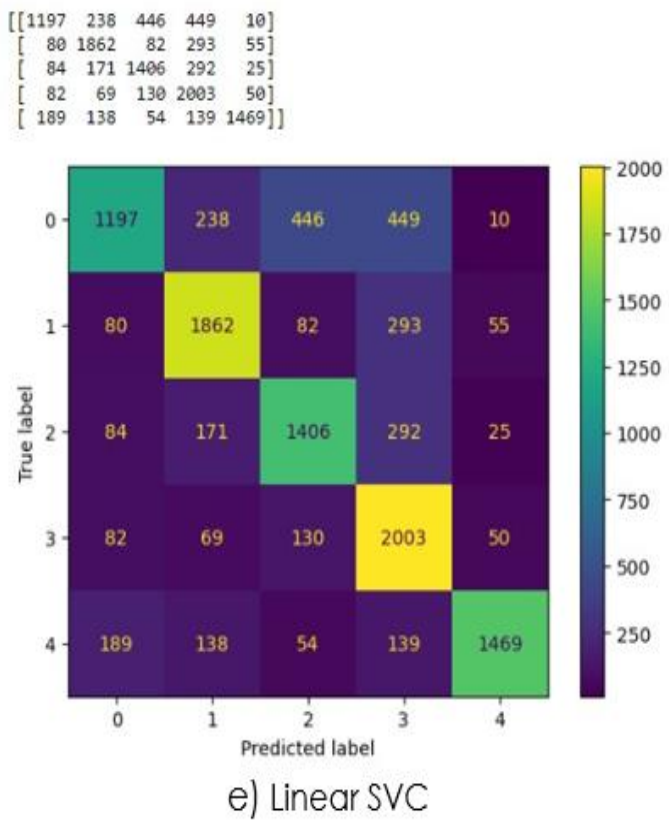


Figure 4.15: Confusion Matrices for Multi-class Classification Models According.

To the figures above, the Random Forest Classifier exhibited inaccuracies in its predictions. It mistakenly identified 82 web addresses as Benign (0) when they actually belonged to different categories. Additionally, it misclassified 39 instances as Defacement (1), 11 instances as Malware (2), 183 instances as Phishing (3), and 16 instances as Spam (4).

The Decision Tree Classifier exhibited inaccuracies in its predictive performance. Specifically, it made mistakes in classifying web addresses into different categories. It wrongly identified 194 instances as Benign when they actually belonged to other categories. Additionally, it made errors by classifying 164 instances as Defacement, 145 instances as Malware, 267 instances as Phishing, and 160 instances as Spam when these instances should have been categorized differently.

The K-Neighbors Classifier incorrectly labeled 190 instances as Benign, 224 as Defacement, 175 as Malware, 222 as Phishing, and 119 as Spam, leading to false positive errors.

Similarly, the Logistic Regression model made prediction errors, misclassifying 526 web addresses as Benign, 435 as Defacement, 376 as Malware, 542 as Phishing, and 244 as Spam.

The Linear Support Vector Classifier, a machine learning algorithm, made mistakes when classifying web addresses into different categories. Specifically, it wrongly identified 435 web addresses as non-harmful (Benign), 616 as Defacement, 712 as Malware, 1173 as Phishing, and 140 as Spam.

Table 4.4 to Table 4.8 show the classification reports for multi-class classification for each model. These reports encompass key metrics such as precision, recall, and F1 score, providing a comprehensive assessment of the classifiers' performance.

Label	Precision	Recall	F1-score
Benign (0)	0.97	0.98	0.97
Defacement (1)	0.98	0.97	0.98
Malware (2)	0.99	0.97	0.98
Phishing (3)	0.93	0.97	0.95
Spam (4)	0.99	0.96	0.97

Table 4.4: Multi-class Classification Report of Random Forest Classifier

Label	Precision	Recall	F1-score
Benign (0)	0.92	0.93	0.92
Defacement (1)	0.91	0.95	0.93
Malware (2)	0.91	0.93	0.92
Phishing (3)	0.90	0.85	0.98
Spam (4)	0.94	0.93	0.93

Table 4.5: Multi-class Classification Report of Decision Tree Classifier

Label	Precision	Recall	F1-score
Benign (0)	0.73	0.51	0.60
Defacement (1)	0.75	0.78	0.77
Malware (2)	0.66	0.71	0.69
Phishing (3)	0.63	0.86	0.73
Spam (4)	0.91	0.74	0.82

Table 4.6: Multi-class Classification Report of K-Neighbors Classifier

Label	Precision	Recall	F1-score
Benign (0)	0.92	0.92	0.92
Defacement (1)	0.93	0.94	0.94
Malware (2)	0.93	0.96	0.94
Phishing (3)	0.88	0.83	0.85
Spam (4)	0.92	0.93	0.93

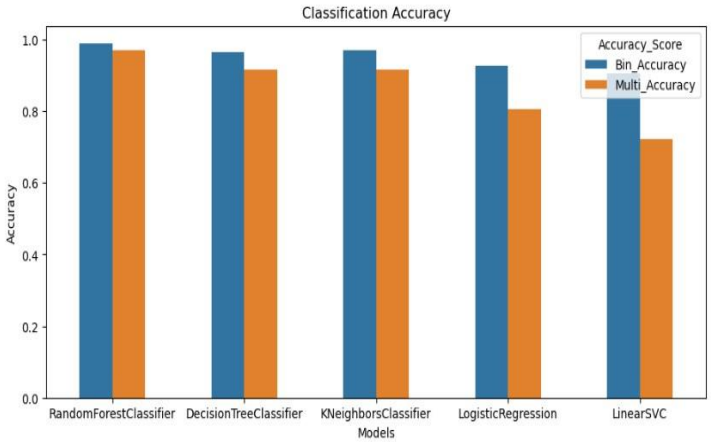
Table 4.7: Multi-class Classification Report of Logistic Regression

Label	Precision	Recall	F1-score
Benign (0)	0.80	0.88	0.84
Defacement (1)	0.82	0.81	0.82
Malware (2)	0.76	0.62	0.68
Phishing (3)	0.78	0.81	0.79
Spam (4)	0.88	0.89	0.89

Table 4.8: Multi-class Classification Report of Linear Support Vector Classifier

DISCUSSION OF RESULTS

The figure above illustrates the accuracy scores achieved by the models in both Binary and Multi-class classification tasks. These models were trained after the application of a dimensionality reduction method known as Principal Component Analysis (PCA) on the dataset.



For the binary classification task, all the machine learning models used performed effectively. However, the Random Forest Classifier outperformed the other models by achieving remarkable results. It achieved an impressive accuracy of 98.9%, indicating it correctly classified nearly all instances. Additionally, it demonstrated a high level of precision at 98.8%, suggesting that it was highly accurate in identifying positive cases while minimizing false positives. The F1 score, which combines precision and recall, was an impressive 99.3%, showing its ability to make accurate positive predictions and effectively balance false positives and false negatives. Furthermore, the Random Forest Classifier exhibited an extremely high true positive rate of 99.7%, meaning it successfully identified a large proportion of actual positive cases. Its true negative rate of 95.6% indicated its reliability in correctly classifying negative cases. The AUC ROC value, which measures the overall performance of the classifier, was 0.977, further highlighting its robustness and effectiveness.

The K-Neighbors Classifier also performed admirably, although it trailed slightly behind the Random Forest Classifier. The Classifier achieved a solid accuracy of 96.9%, showcasing its proficiency in classifying data points correctly. It also exhibited a respectable precision of 97.8%, and a commendable F1 score of 98.1%. Additionally, the K-Neighbors Classifier displayed a noteworthy true positive rate of 98.4%, indicating its ability to accurately identify actual positive cases. Its true negative rate, while slightly lower at 91.6%, still showed its ability to reliably classify negative cases. The AUC ROC score of 0.950 further confirmed its overall impressive performance in distinguishing between the two classes.

The Decision Tree Classifier performed exceptionally well with an accuracy of 96.4%, highlighting its effectiveness in classifying data accurately. It also displayed outstanding precision at 97.6% and a strong F1 score of 97.7%, indicating its ability to make precise positive predictions. Furthermore, it had high true positive and true negative rates of 97.8% and

91.2%, respectively, which further emphasized its reliability. Additionally, the classifier achieved an AUC ROC score of 0.945, indicating its overall effectiveness.

On the other hand, the Logistic Regression model achieved a commendable accuracy of 92.6%, which is quite good, although slightly lower than the previous models. It demonstrated a high precision of 95.4% and an F1 score of 95.3%, signifying its capability to make accurate positive predictions. Its true positive rate of 95.2% and true negative rate of 82.9% are also acceptable. Additionally, it obtained an AUC ROC score of 0.890.

The Linear Support Vector Classifier performed with the lowest overall accuracy at 90.6%. However, it showcased a noteworthy level of precision at 94.2% and achieved an F1 score of 94.0%, indicating its ability to make accurate positive predictions. On the flip side, it had comparatively lower rates of true positives
(93.9%) and true negatives (78.5%) when compared to the other models. Additionally, it obtained the smallest AUC ROC score, measuring 0.862.

Based on the findings presented above, it is advisable to employ the Random Forest Classifier to distinguish between safe and malicious URLs. This classifier has demonstrated remarkable performance across various evaluation criteria. It also had the lowest false prediction rate. This outstanding capability positions the Random Forest Classifier as an effective tool for detecting malicious URLs. For multi-class classification, the Random Forest Classifier achieved the highest accuracy of 97.0%, indicating that it was able to make accurate predictions for the majority of the cases in the dataset it was tested on. This suggests that it's a robust model for this specific task.

Both the Decision Tree Classifier and K-Neighbors Classifier achieved an accuracy of 91.6%. This implies that these two models perform similarly and are quite reliable in making correct predictions.
The Logistic Regression model achieved an accuracy of 80.7%, indicating that it is moderately accurate but not as strong as the Random Forest or Decision Tree models.

The Linear Support Vector Classifier had the lowest accuracy, with only 72.1% correct predictions. This suggests that it may not be the most suitable choice for this particular task.
The multi-class classification reports highlight the Random Forest Classifier's exceptional performance in precisely classifying diverse instances of distinct types of Malicious URLs. The classifier attained high precision, recall, and F1-scores, between 95% and 99%, which indicates that the classifier is effective in distinguishing between the classes.
The K-Neighbors Classifier and the Decision Tree Classifier also achieved good precision, recall, and F1-scores across all classes, indicating a balanced and accurate classification for each category.

The Logistic Regression model performed reasonably well for some classes, such as Benign, Defacement and Spam, with F1-scores of around 0.84, 0.82 and 0.89 respectively. However, it struggled with the Malware class, where the F1-score is only 0.68.

The Linear Support Vector Classifier had challenges in achieving high recall for most classes, particularly Benign and Malware. It also had relatively lower precision for Benign as well.

Hence, in the context of this multi-class classification problem, the Random Forest Classifier demonstrates superior performance by effectively assigning the correct class labels to most of the data points. This remarkable performance highlights the Random Forest Classifier as a promising choice for reliably and accurately identifying various forms of malicious web addresses.

In summary, this study recommends the Random Forest Classifier as the most suitable model for the detection and prediction of malicious URLs in both binary and multi-class classification context. Whether the goal is to ascertain if a given web address is malicious, or to categorize detected malicious URLs into distinct groups, the Random Forest Classifier is the preferred choice due to its exceptional performance in terms of accuracy, precision, recall, and F1 scores for this particular classification task.

5. RESEARCH SUMMARY

The paper "Prediction and Detection of Malicious URLs using Machine Learning" highlights the importance of web security and the need for reliable methods to detect rogue URLs. The literature review provides a thorough analysis of previous studies and methods, laying the groundwork for the research. The study design is explained, including the selection of machine learning methods for binary and multi-class classification. The Random Forest Classifier is highlighted as the best option for predicting and detecting dangerous URLs.

The results present data on machine learning models' performance in binary and multi-class classification problems. The Random Forest Classifier is highlighted as the best method, performing the binary classification test with an excellent accuracy of 98.9%. Its F1 score of 99.3% and precision of 98.8% demonstrate its effectiveness in distinguishing between trusted and harmful URLs. The AUC ROC score of 0.977 further supports its status as the best model for accurate and reliable detection of dangerous URLs.

6. CONCLUSION

This research concludes with important gains for both theoretical computer security research and real-world web protection. The significance of the Random Forest Classifier in accurately identifying and categorizing harmful URLs is highlighted by the study's findings, in particular the classifier's continuously remarkable performance in both binary and multi-class classification tasks. While this study addresses an important current need, it also contributes to a deeper understanding of how machine learning algorithms might be deployed to counteract the ever-evolving threats that plague the digital sphere. Recommending the Random Forest Classifier as the best model for detecting fraudulent URLs has significant real-world ramifications. A practical, implementable answer is provided, which has the potential to boost the efficiency with which security systems can detect and counteract online threats. This study adds to the growing body of work aimed at improving cybersecurity procedures and so reinforcing the online world against intrusion. The results of this research have the potential to benefit both the academic community and the cybersecurity industry. The former will benefit from a deeper understanding of machine learning's application to cybersecurity, while the latter will benefit from a more robust and versatile tool to protect the web from threats. More than just an intellectual exercise, this study can serve as a guidepost for improving cyber defences in today's interconnected world.

7. CONTRIBUTION TO KNOWLEDGE

This study demonstrates the improved performance of the Random Forest Classifier in predicting and detecting harmful URLs, contributing to cybersecurity and machine learning. It enhances our understanding of how machine learning can combat digital threats and improve web security. The research also contributes to the growing body of work on explainable artificial intelligence (XAI) in cybersecurity, highlighting the need for more open models. The study also advances the field of AI by demonstrating the flexibility and scalability of machine learning algorithms in dynamic situations. It also contributes to the ongoing discussion on responsible AI development and deployment.

8. FUTURE WORK

Future research should explore ensemble learning techniques, deep learning models like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), and real-time monitoring systems to improve malicious URL prediction and detection. Collaboration with cybersecurity professionals and organizations is crucial for deploying these models. Future research should focus on updating and retraining models to anticipate new attack pathways. More comprehensive datasets covering a wider range of malicious URL categories and trends are needed for model generalization. Advancements in explainable AI (XAI) techniques can enhance model interpretability and openness to scrutiny. Collaboration with experts in machine learning, cybersecurity, and web development is essential for developing solutions that consider not only detection but also remediation and mitigation strategies.

9. REFERENCES

[1] Adebowale, M.A., Lwin, K.T. and Hossain, M.A. (2020) 'Intelligent phishing detection scheme using deep learning algorithms,' Journal of Enterprise Information Management, 36(3), pp. 747–766. https://doi.org/10.1108/jeim-01-2020-0036.

[2] Alomari, E., Nuiaa, R.R., Alyasseri, Z.A.A., Mohammed, H.J., Sani, N.S., Esa, M.I.

and Musawi, B.A. (2023) 'Malware detection using Deep Learning and Correlation-Based feature selection,' Symmetry, 15(1), p. 123. https://doi.org/10.3390/sym15010123.
[3] Anderson, J. (2020). Leveraging Machine Learning for Malicious URL Detection. In Proceedings of the International Conference on Cybersecurity and Computer Forensics (ICCCF).
[4] Anitha, T. et al. (2023) 'A novel methodology for malicious traffic detection in smart devices using BI LSTM–DT-dependent deep learning methodology,' Neural Computing and Applications, 35(27), pp. 20319–20338. https://doi.org/10.1007/s00521-023-08818-0.
[5] Alzubi, O.A., Qiqieh, I. and Alzubi, J.A. (2022) 'Fusion of deep learning based cyberattack detection and classification model for intelligent systems,' Cluster Computing, 26(2), pp. 1363–1374. https://doi.org/10.1007/s10586-022-03686-0.
[6] Brown, A. (2017). Machine learning approaches malicious URL detection. In Proceedings of the International Conference on Cybersecurity (ICCS) (pp. 45-56).
[7] Bhuyan, M. H., Bhattacharyya, D. K., Kalita, J. K., & Sarmah, P. (2020). Explainable AI for Cybersecurity: A Review. IEEE Access, 8, 120089-120111.
[8] Brown, A., Martinez, L., & Garcia, M. (2018). Machine learning models for realtime malicious URL detection. Journal of Network Security, 22(5), 567-580.
[9] Chen, Q., Martinez, L., & Garcia, M. (2019). Comprehensive evaluation metrics for malicious URL detection. Journal of Information Assurance and Security, 14(3), 234-249.
[10] Feng, Y., Wang, Q., Zhou, Z., & Li, Y. (2020). Explainable AI in Cybersecurity. IEEE Access, 8, 182494-182508.
[11] Gopinath, M. and Sethuraman, S.C. (2023) 'A comprehensive survey on deep learning based. malware detection techniques,' Computer Science Review, 47, p. 100529. https://doi.org/10.1016/j.cosrev.2022.100529.
[12] Garcia, M., Johnson, R., & Smith, J. (2020). Feature selection and engineering for enhancing. malicious URL detection. Journal of Information Security Research, 25(1), 45-58. [13] Garcia, M., & Martinez, L. (2018). Ethical considerations in cybersecurity research. Journal of Cyber Ethics, 11(1), 78-91.
[14] Jones, P., & Brown, A. (2019). Enhancing cybersecurity with machine learning algorithms.
International Journal of Information Security, 17(4), 423-438.
[15] Johnson, R., Smith, J., & Brown, A. (2016). Ineffectiveness of traditional methods in detecting malicious URLs. Cybersecurity Journal, 8(4), 321-335.
[16] Keerthi Vasan, K., & Surendiran, B. (2016). Dimensionality reduction using Principal Component Analysis for network intrusion detection. Perspectives in Science, 8, 510–512. https://doi.org/10.1016/j.pisc.2016.05.010
[17] Liu, X., Hu, L., Kang, J., & Ma, J. (2019). Research on Malicious URL Detection Based on Machine Learning Algorithms. In Proceedings of the International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC).
[18] Martinez, L. (2019). Exploring the application of supervised learning in malicious URL detection. In Proceedings of the International Conference on Cybersecurity and Privacy (ICCP) (pp. 87-98).
[19] Matsuzaka, Y., & Uesawa, Y. (2023). Ensemble learning, Deep Learning-Based and molecular Descriptor Based quantitative Structure–Activity relationships. Molecules, 28(5), 2410. https://doi.org/10.3390/molecules28052410
[20] Naim, O., Cohen, D. and Ben-Gal, I. (2023) 'Malicious website identification using design attribute learning,' International Journal of Information Security, 22(5), pp. 1207–1217. https://doi.org/10.1007/s10207-023-00686-y.
[21] Prabakaran, M.K., Chandrasekar, A.D. and Sundaram, P.M. (2023) 'An enhanced deep learning-based phishing detection mechanism to effectively identify malicious URLs using variational autoencoders,' Iet Information Security, 17(3), pp. 423–440. https://doi.org/10.1049/ise2.12106.
[22] Patgiri, R., Biswas, A. and Nayak, S. (2023b) 'deepBF: Malicious URL detection using learned Bloom Filter and evolutionary deep learning,' Computer Communications, 200, pp. 30-41. https://doi.org/10.1016/j.comcom.2022.12.027.
[23] Patgiri, R., Katari, H., Kumar, R., Sharma, D. (2019). Empirical Study on Malicious URL Detection Using Machine Learning. In: Fahrnberger, G., Gopinathan, S., Parida, L. (eds) Distributed Computing and Internet Technology. ICDCIT 2019. Lecture Notes in Computer Science (), vol 11319. Springer, Cham. https://doi.org/10.1007/978-3030-05366-6_31.
[24] Reddy, G. T., Reddy, M. P. K., Lakshmanna, K., Kaluri, R., Rajput, D. S., Srivastava, G., & Baker, T. (2020). Analysis of Dimensionality Reduction Techniques on Big Data. IEEE Access, 8, 54776–54788. https://doi.org/10.1109/ACCESS.2020.2980942
[25] Ramanauskaite, S., & Garsva, G. (2019). Malicious URL Detection Based on Machine Learning Algorithms: A Comparative Study. In Proceedings of the International Conference on Computational Intelligence in Security for Information Systems (CISIS).

[26] 'The Significance of machine learning and deep learning techniques in Cybersecurity: A Comprehensive review' (2023) Iraqi Journal for Computer Science and Mathematics, pp. 87–101. https://doi.org/10.52866/ijcsm.2023.01.01.008. [27] Sahoo, D. (2017) Malicious URL Detection using Machine Learning: A Survey. https://arxiv.org/abs/1701.07179.
[28] Sharma, A., Sharma, P. K., & Gaur, M. S. (2018). A Framework for Real-Time Malicious URLDetection. In Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon).
[29] Sahu, S. K., Mokhade, A., & Bokde, N. D. (2023). An Overview of Machine learning, deep learning, and Reinforcement Learning-Based Techniques in Quantitative Finance: Recent progress and challenges. Applied Sciences, 13(3), 1956. https://doi.org/10.3390/app13031956
[30] Sudeep Tanwar, Ramani, T., & Tyagi, S. (2017). Dimensionality Reduction Using PCA and SVD in Big Data: A Comparative Case Study. Springer EBooks, 116–125. https://doi.org/10.1007/978-3-319-73712-6_12
[31] Smith, J., & Brown, A. (2018). Signature-based antivirus software for malicious URL detection. Journal of Cybersecurity, 5(2), 123-135.
[32] Folorunsho F., Korkor M., *et al.* (2023). Quantitative approaches to forecasting the economic impact of technological disruptions in making informed decisions for sustainable economic growth in the U.S, *International Journal of Advance Research, Ideas, and Innovations in Technology,* 9(6), 193-201.
[33] Smith, J., Johnson, R., & Garcia, M. (2020). Predicting and detecting malicious URLs using machine learning. Journal of Computer Security, 28(3), 312-326.
[34] Smith, J., & Johnson, R. (2017). Ethical considerations in machine learning for cybersecurity. Journal of Cybersecurity Ethics, 14(2), 189-203.
[35] Vanhoenshoven, G. Nápoles, R. Falcon, K. Vanhoof and M. Köppen, "Detecting malicious URLs using machine learning techniques," 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 2016, pp. 1-8, doi: 10.1109/SSCI.2016.7850079.
[36] Verma, S., Chugh, S., Ghosh, S., & Rahman, B. M. A. (2023). A comprehensive deep learning method for empirical spectral prediction and its quantitative validation of nanostructured dimers. Scientific Reports, 13(1). https://doi.org/10.1038/s41598-02328076-3
[37] Xiao, G., Zhu, B., Zhang, Y., & Gao, H. (2023). FCSNet: A quantitative explanation method for surface scratch defects during belt grinding based on deep learning. Computers in Industry, 144, 103793. https://doi.org/10.1016/j.compind.2022.103793

