Phishing Website Detection Using Data Mining Techniques

Prudhvi Chowdary Chirumamilla, Raga Poojitha Botlagunta, Rohan Naga Venkata Mayukh Ungarala
Department of Applied Data Science, San Jose State University (SJSU)
Dr. Shayan Shams
Data 240: Data Mining and Analytics

I. Motivation and Background Information

Phishing is a kind of cyberattack in which perpetrators deceive targets into divulging private information, such as bank account login passwords, by means of social engineering. These phishing attempts have the potential to cause serious harm, such as identity theft, money loss, and reputational harm. Even with a variety of defenses against phishing, it might be difficult to identify malicious URLs due to the growing complexity of attackers. By examining their traits and trends, data mining technologies are frequently used to identify phishing websites. These programs categorize newly created websites using algorithms that compare and contrast elements of legitimate and phishing websites. However, the current approaches frequently have large false positive rates and poor accuracy. This emphasizes the need for more accurate and practical methods of spotting phishing websites. The purpose of this project is to create a system for precise phishing site detection via data mining techniques. In order to complete this project, a sizable and varied dataset of authentic and fraudulent websites will be gathered. Their properties will then be extracted, and machine learning techniques will be used for analysis and classification. The ultimate outcome will be a system that can accurately detect phishing websites, strengthening defenses against malicious URLs and phishing attempts.

II. Literature Review

The full literature analysis of several techniques for spotting phishing websites is provided in the research paper "Prediction of Phishing Website for Data Security Using Various Machine Learning Algorithms" published in 2021. These include of HTML features, image- and visual-based techniques, and domain whitelisting and blacklisting tactics. This work stands out for concentrating on URL attributes to differentiate phishing sites, using information from Kaggle and the UCI Machine Learning Repository. Li et al. (2020) presented a unique approach that combines machine learning techniques and graph theory for the identification of phishing websites. This approach entails researching the characteristics of phishing sites and creating a graph model to show how they are related to one another. Then, using machine learning techniques, new websites are classified using this graph model, with an astounding 96.3% accuracy, demonstrating the promise of graph theory as a major help in phishing detection. Phishing assaults are getting increasingly common, and there are several methods available to identify them. One popular strategy is to use data mining techniques to detect unique patterns by examining the features of phishing websites. In order to identify phishing sites, Singh et al. (2021) suggested a method that makes use of data mining algorithms to examine URL characteristics and website content. With a 98% success rate in identifying phishing sites, this method proved to be highly effective, demonstrating the usefulness of data mining in thwarting phishing attacks. "Detecting phishing websites using machine learning technique (2021)" is the title of the study. There is an exponential rise in the number of victims as a result of ineffective security solutions. Previous studies indicate that the phishing detection system's performance is constrained. To safeguard users from cyberattacks, a clever strategy is required. The author of this paper suggested

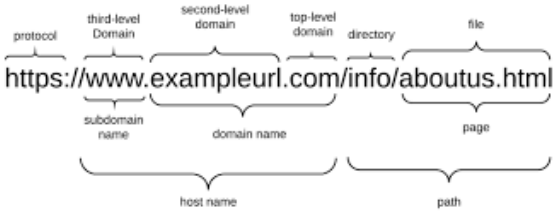
a machine learning-based URL detection method. The technique of recurrent neural networks is used to identify phishing URLs. Researchers used 5800 genuine and 7900 malicious websites to test the suggested strategy. The results of the studies demonstrate that the suggested strategy outperforms more contemporary techniques for malicious URL detection. Kshirsagar and Reddy (2019) used decision trees and random forest algorithms to create a solution for detecting phishing websites. To classify new websites, our system examined a set of characteristics from both authentic and phishing websites. Their results show that decision trees and random forest algorithms are successful in identifying phishing websites, with a 92.6% accuracy rate. Other than these research in the literature review, we have implemented 6 different models int his project and secured above 80% for all the models by training our data.

III. Methodology

Data Collection: Prior to starting the project's data collecting, a number of important factors needed to be considered, including the project's goal, the kinds of data required, and how they should be presented. The plan for gathering data was the most important component. This project's Kaggle dataset was gathered from a variety of sources, including web crawls, open malware databases, and blacklists. Preprocessing of this dataset was done to eliminate unnecessary and redundant URLs and classify them as safe or harmful. To determine the most prevalent top-level domains (TLDs) and subdomains in the dataset, a domain analysis was also carried out. The data, which included 651,191 URLs in total, was obtained straight from Kaggle. Figure 3 shows the distribution of these URLs, which comprise 32,520 harmful URLs, 428,103 benign or safe URLs, 96,457 changed URLs, and 94,111 phishing URLs. An effective dataset selection is essential to a data mining project's success. This specific dataset combines information from five sources. Although the main source was Kaggle, the URLs were first gathered from several sources, put together into a separate data frame, and then combined, keeping only the URLs and their categorization. As shown in the preceding figure, a URL normally comprises of the hostname, pathways, web address port, and top-level domain.

Data Preprocessing: Cleaning and filtering the dataset during data pre-processing ensures its correctness and dependability. Managing missing values, getting rid of duplicate entries, and dealing with outliers and anomalies are all part of this step. Exploratory data analysis (EDA) need to be the first stage of every data analysis effort. EDA is going over a dataset to find trends and abnormalities before generating theories based on the information. Intriguing correlations between variables can also be found using it, as can patterns, outliers, and other anomalies in the data. The main technique for obtaining pertinent attributes from the URLs was feature engineering. Important characteristics found were the length of the URL, if special characters were present, and how frequently certain phrases appeared. Then, using methods like one-hot encoding, categorical characteristics were transformed into numerical representations that worked with machine learning models. The Mutual Information (MI) scores for each instance were shown in a bar chart. It was clear from this that HTTPS had the best score, followed by AnchorURL, and that UsingPopWindow had the lowest MI value.

Figure 1: Anatomy of Web address



The frequency of various URL shortening services in the dataset is with the x- and y-axes representing URL shortening services' predominance. It demonstrates that not a single URL shortening tool was used to

Figure 2: Image of data after preprocessing

	url	type	Category	url_len	domain	@	?	-	=	.	...	!	*	,	//	abnormal_url	https	digits_in_url	letters_in_url	Shortining_Service	having_ip_address
0	br-icloud.com.br	phishing	2	16	br-icloud.com.br	0	0	1	0	2	...	0	0	0	0	0	0	0	13	0	0
1	mp3raid.com/music/krizz_kaliko.html	benign	0	35	mp3raid.com	0	0	0	0	2	...	0	0	0	0	0	0	1	29	0	0
2	bopsecrets.org/rexroth/cr/1.htm	benign	0	31	bopsecrets.org	0	0	0	0	2	...	0	0	0	0	0	0	1	25	0	0
3	http://garage-pirenne.be/index.php?option=com_...	defacement	1	84	garage-pirenne.be	0	1	1	4	2	...	0	0	0	1	1	0	7	60	0	0
4	http://adventure-nicaragua.net/index.php?optio...	defacement	1	235	adventure-nicaragua.net	0	1	1	3	2	...	0	0	0	1	1	0	22	199	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
651186	xbox360.ign.com/objects/850/850402.html	phishing	2	39	xbox360.ign.com	0	0	0	0	3	...	0	0	0	0	0	0	12	21	0	0
651187	games.teamxbox.com/xbox-360/1860/Dead-Space/	phishing	2	44	games.teamxbox.com	0	0	2	0	2	...	0	0	0	0	0	0	7	29	1	0
651188	gamespot.com/xbox360/action/deadspace/	phishing	2	38	gamespot.com	0	0	0	0	1	...	0	0	0	0	0	0	3	30	1	0
651189	en.wikipedia.org/wiki/Dead_Space_(video_game)	phishing	2	45	en.wikipedia.org	0	0	0	0	2	...	0	0	0	0	0	0	0	36	0	0
651190	angelfire.com/goth/devilmaycrytonite/	phishing	2	37	angelfire.com	0	0	0	0	1	...	0	0	0	0	0	0	0	33	0	0

651191 rows × 24 columns

shorten the bulk of the dataset's URLs. "Bit.ly" was the most widely utilized service among those that were shortened. The following Heat Map suggests multicollinearity and suggests that the high correlation between many factors may make the categorization approach less successful. As part of the data preparation process, unnecessary data has to be eliminated in order to resolve incomplete and missing data. Because the dataset came from Kaggle, it was guaranteed to be comparatively clean and free of significant inconsistencies. The user didn't have any gaps or missing fields to fill in. As a result, no extra steps were required to deal with missing data.

Figure 3: Bar plot for the MI score

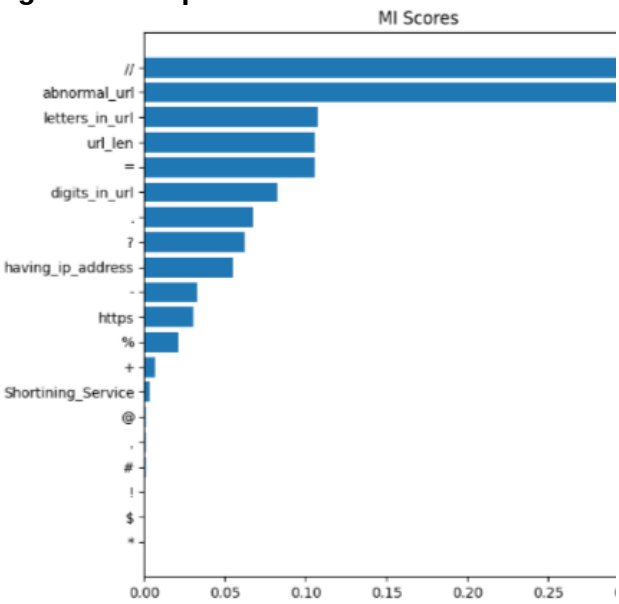
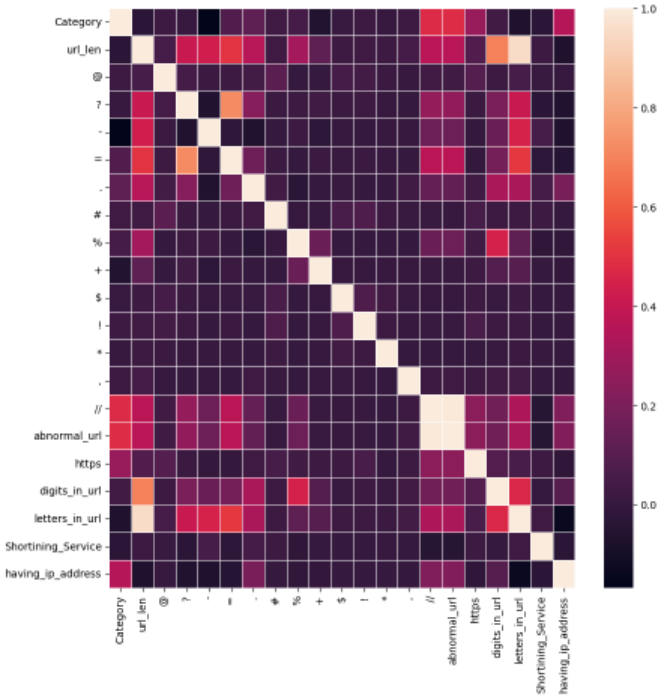


Figure 4: Heatmap



Data Preparation: In order to enable data scientists and analysts to extract insights and forecast future trends, raw data is converted into a format that is appropriate for machine learning algorithms during the data preparation process. This procedure entails removing any errors and thoroughly cleaning the data before feeding it into deep learning or machine learning algorithms. Partitioning the data is a crucial machine learning technique that avoids overfitting, a phenomenon in which a model that is too closely matched to its training set of data becomes unusable with fresh data. The

dataset was split into training and testing subsets in order to address this. For this division, the popular split ratio of 80:20 was selected. To make sure the model was accurate and effective after training, its performance was assessed by contrasting the training and test sets. For splitting the dataset, we have also tried by using K-fold cross validation method, but it is taking more time than the traditional splitting method but the outcome is same for both.

Models Proposal: We have proposed 5 machine learning models in this project along with one method in machine learning, we just want to try different models which has different aspects. They are Gaussian Naïve Bayes, AdaBoost classifier, Decision tree, Random Forest, K-Nearest neighbors and the method is Stochastic Gradient Descent.
Multiple decision trees are used in the Random Forest algorithm, an ensemble learning technique, to generate predictions. It is well known for efficiently managing noisy and high-dimensional data, and it works particularly well for detecting fraudulent URLs. Using the preprocessed dataset and the chosen features as input, this algorithm will be trained. Decision trees, which divide the data into subsets according to the most important features recursively, provide a straightforward yet effective method for categorization. The preprocessed dataset will be used to train the suggested Decision Tree model, which will use the discovered features as its input. Another ensemble learning technique called AdaBoost combines many weaker models to create a robust one. Less emphasis is placed on cases that were correctly classified and more on those that were misclassified. The preprocessed dataset containing the collected features will be used to build the AdaBoost model in this investigation. KNN (K-Nearest Neighbors) is a simple yet effective method that works well for classification applications. By determining an instance's k-nearest neighbors and allocating the most common class among them, it may be classified. To do this, the suggested KNN model will be trained using the preprocessed dataset and the features that were extracted. In order to reduce dimensionality and ultimately the number of features in a dataset, one strategy for doing so is called stochastic gradient descent, or SGD. It functions by splitting the data into many orthogonal components, each of which represents a distinct degree of variability in the data. The preprocessed dataset and the detected features will be used by the suggested SGD model. Finally, Gaussian distributions are used to describe the variability of each class and calculate the likelihood of each class for a given occurrence in the Gaussian NB (Naive Bayes) method, which is efficient in classification tasks. Using the retrieved features as input, the preprocessed dataset will be used to train the Gaussian NB model.

IV. Model validation, Evaluation and Results of the Experiments

After training six different models, criteria including accuracy, F1 score, precision, and recall were used to evaluate each model's performance.
Accuracy: This statistic calculates the percentage of accurate predictions a model makes relative to all forecasts. It is determined by dividing the total number of forecasts by the number of accurate guesses. A simple measure of the overall performance of the model is accuracy.
F1-score – Precision and recall are harmonic means that make up the F1 Score. It is especially helpful when there is an unequal class distribution and you need to strike a balance between precision and recall.
Precision–The precision of positive forecasts is represented by their accuracy. It is calculated as the ratio of all expected positive predictions to genuine positive forecasts. This measure is essential when the expense of a false positive is substantial.
Recall - Recall, which is often referred to as Sensitivity or True Positive Rate, quantifies the percentage of real positives that are accurately detected. It is calculated as the

ratio of real positives to the total of false negatives and true positives.
All models and method implemented in this project achieved good accuracy. But three models performed best among these 6. They are K-Nearest neighbor, Random Forest and Decision Tree. These criteria serve as a foundation for assessing the efficacy of each model, especially when it comes to identifying phishing websites.

We have achieved the results, we also performed test evaluation by taking some other website links randomly which are not in our dataset and did the same method from the start and checked out the result. Surprisingly, we have got the same results. So, we considered our project is effective. A comparative examination of all the models that were implemented is shown in the table below.

Table 1: Results of the experiments

Models	**Accuracy**	**F1-Score**	**Recall**	**Precision**
Naive Bayes	0.79	0.88	0.92	0.85
Stochastic Gradient Descent	0.81	0.89	0.94	0.85
AdaBoost Classifier	0.82	0.90	0.98	0.84
K-Nearest Neighbor	0.89	0.93	0.96	0.91
Decision Tree	0.90	0.94	0.97	0.92
Random Forest	**0.91**	**0.95**	**0.98**	**0.92**

V. Discussion and Future Improvements

Creating new strategies is crucial as phishing website techniques advance and becoming more complex. As new data becomes available, it's critical to adaptively incorporate new feature extraction techniques to preserve the model's accuracy and relevance. By using graph database approaches, hidden relationships and trends between different users of phishing websites might be uncovered. This might entail investigating novel techniques for feature extraction, putting more sophisticated machine learning algorithms into practice, or incorporating different types of data analysis. Retraining and frequent updates of the model can help achieve this. To summarize, this study used a variety of machine learning methods, including AdaBoost, SVD, k-nearest neighbors, decision trees, random forests, and Gaussian naive Bayes, to identify phishing websites. The efficacy of these machine learning models in precisely detecting phishing websites was demonstrated by their accuracy, which ranged from 79% to 91%. But there are restrictions on

the project. It's possible that the modest size of the dataset utilized may not accurately reflect the an impact on how well the machine learning algorithms work. In order to verify the predictions on larger and more varied datasets and investigate different feature selection strategies, more study is required. It's critical to create new approaches for identifying and thwarting phishing website assaults as they continue to change. It is necessary to dynamically include new feature extraction techniques as fresh data becomes available in order to maintain the model current and correct. This might mean investigating novel feature extraction methods, using more advanced machine learning algorithms, or combining several forms of data analysis. The model must be continuously updated and retrained in order to achieve this goal.

Community Contribution: The cost of cybersecurity breaches is rising, both in terms of money and effort. These assaults can take many different forms, such as ransomware, malware, and phishing via emails or websites. A cybersecurity team plays a critical function in a company. Businesses are working harder to educate staff members about the importance of cybersecurity. Among the strategies are lunchtime lectures, regular updates, and similar events. Every month, employees must participate in online training to increase their knowledge of cybersecurity hazards. Because our technology is built to identify harmful websites, it may help raise awareness.

VI. References

Kshirsagar, R., & Reddy, P. V. S. (2019). Detecting phishing websites using decision trees and random forest algorithms. International Journal of Computer Science and Mobile Computing, 8(3), 7-15.

J. Kumr, A. Santhanavijayan, B. Janet, B. Rajenran and B. S. Bindhumadhava, "Phishing Website Classification and Detection Using MachineLearning," *2020International Conference on Computer Communication and Informatics (ICCCI)*, Coimbatore, India, 2020, doi: 10.1109/ICCCI48352.2020.9104161.

Singh, S., Kumar, A., & Sharma, P. (2021). Phishing website detection using data mining techniques. International Journal of Advanced Computer Science and Applications, 12(2), 455-460

Park, Andrew J., Ruhi Naaz Quadari, and Herbert H. Tsang. "Phishing website detection framework through web scraping and data mining." 2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2017.

Li, Y., Xu, Z., Wang, W., & Liu, X. (2020). Phishing website detection using machine learning and graph theory. Security and Communication Networks, 2020, 1-11.

Ahammad, SK Hasane, et al. "Phishing URL detection using machine learning methods." Advances in Engineering Software 173 (2022): 103288.

Das Guptta, Sumitra, et al. "Modeling hybrid feature-based phishing websites detection using machine learning techniques." Annals of Data Science (2022): 1-26.

Roy, Sanjiban Sekhar, et al. "Multimodel phishing url detection using lstm, bidirectional lstm, and gru models." Future Internet 14.11 (2022): 340.