Classifying Fraudulent Websites with Machine Learning Algorithms

Gulay Cicek[1], and Efe Genişoğlu[2]
[1,2]Department of Software Engineering, Faculty of Engineering Architecture
Istanbul Beykent University, Sariyer, Istanbul, Turkey
[1]gulaycicek@beykent.edu.tr, [2]2203013061@student.beykent.edu.tr

***Abstract*—**The rapid spread of the Internet has brought many conveniences to daily life, while also increasing security problems. In recent years, phishing attacks have become the most common of these problems. Phishing attacks aim to obtain personal information about users via a fake website or an e-mail. It is very important that personal information (name, surname, credit card information, address, etc.) provided while making transactions on websites does not fall into the hands of malicious individuals. In this study, it is aimed to classify fake websites using URL analysis using machine learning algorithms using predetermined features. The dataset used in the study was taken from the Machine Learning Repository (UCI). 13 different features were selected from the dataset consisting of 134,850 legitimate and 100,945 phishing URLs and classified as legitimate or phishing with the specified parameters. 5 different machine learning methods were applied: Random Forest (RF), Support Vector Machines (SVM), Naive Bayes (NB), C4.5 and AdaBoost. Among these algorithms, Random Forest achieved the highest accuracy rate with 99.7%. The method with the lowest accuracy rate was 95.58% Naive Bayes. In order to improve detection, new features that were not already in the dataset, such as the number of special characters and the presence of numbers in the domain name, were added. In the future, a more powerful model can be created by including deep learning methods.

Keywords: Phishing Detection, Fake Sites, Machine Learning, Cyber Security, URL Analysis.

I. Introduction

The Internet is a giant communication network that allows computers around the world to connect and share information. Today, it has become a widespread and powerful tool that provides convenience in every aspect of our lives. According to the ”Internet use in 2024” analysis prepared by DataReportal, the number of people using the Internet in 2024 will exceed 5.35 billion (66.2% of the world’s population).[1] The widespread use of the internet has made it possible for us to do many of the tasks we do in our daily lives remotely via the internet. We no longer need to go to the mall to buy clothes or to the bank when we need to take out a loan. All that is needed is to find the relevant website, enter the necessary information and complete the transaction. So is it safe to enter personal information on every website?

Phishing, or phishing/baiting, is the practice of fraudsters posing as a trusted person or company and deceiving their victims into sharing sensitive information such as passwords and credit card numbers. [2] In this method, fraudsters widely use unsolicited e-mails that appear to come from shopping or banking sites in phishing attacks, copy site codes and designs from official websites one-to-one, and create fake websites that look real on their own websites. Since these sites are designed extremely realistically, if people do not have information about safe internet use, the possibility of being deceived is very high. [3]

When you try to change your email, password or other important information, trusted platforms follow some verification procedures to verify that it is you who made the change. These can be in the form of sending a verification code/link to your email address or phone. [4] If you are not careful enough, there is no turning back after these procedures, you are now a victim. After these steps, your personal information can be used for any purpose. People who get hold of your information can empty your bank accounts, open fake accounts with your identity and commit fraudulent activities. [5] If phishing occurred via malware installation, they may encrypt files on your device and threaten and demand ransom from you. [6] If you are a company employee, they can access the company’s corporate network and data, obtain confidential information, and damage the company’s reputation and financial situation. [7] While all this may sound scary, there are some ways to avoid phishing attacks.

- Don’t click on untrusted links.
- Install antivirus programs on your devices. It will help protect you from malware.
- Check the URLs of the sites. Fake sites use addresses that are similar to the original URLs but with slight differences to trick you. [8]
- Pay attention to the person sending the email. Corporate emails come from companies’ own domain addresses. Fake emails usually use domain names that we often come across, such as Gmail and Hotmail. According to research by the Anti-Phishing Working Group (APWG), in the second quarter of 2024, 72.4% Google and 16.3% Microsoft were used as webmail providers in BEC (Business Email Compromise) attacks. [9] [10]

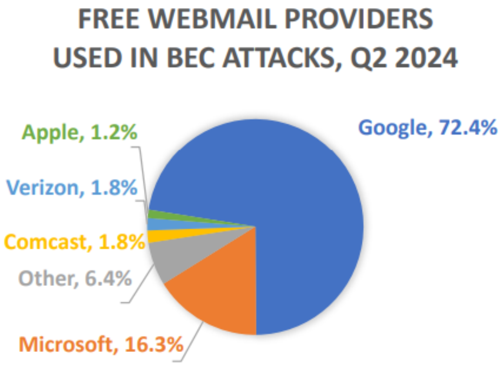

Fig. 1: Statistics showing free webmail providers used in BEC attacks.[9]

- Activate two-step verification on your accounts. Two-step verification will send a verification code to your phone via SMS or email, even if you just want to log in to your account, and confirm that you are the person making the transaction.
- Do not click on shortened URLs. Shortened links like ”goo.gl” can mislead you and direct you to fake sites.
- Change your passwords frequently.
- Avoid using public Wi-Fi networks. Hackers are very likely to infiltrate unencrypted networks found in cafes, airports, and shopping malls. If you absolutely must connect to these networks, use a VPN (Virtual Private Network). Using a VPN can hide your location and block the site/provider’s trackers.[11] [12]
- Use highly secure and complex passwords for each of your accounts. Avoid easy-to-guess passwords like ”123456.”[13]
- Check if the site you are visiting has an SSL (Secure Socket Layer) certificate. SSL is used for secure data communication on websites. This certificate encrypts the data between the web server and the visitor and the user’s information (passwords, credit card information, etc.) is transmitted securely. SSL also verifies the identity of the site, so users are protected from logging into fake sites. You can verify whether the SSL certificate is available by looking at the lock icon in the link section of the page. [14]
- Don’t believe messages that promise you a gift card or a prize. No one will give you anything for free. According to APWG’s research, ”gift card” scams are the leading phishing category at 38.1%. Advance-fee fraud comes next. In this type of scam, you are asked to send money upfront in order to make a big profit.

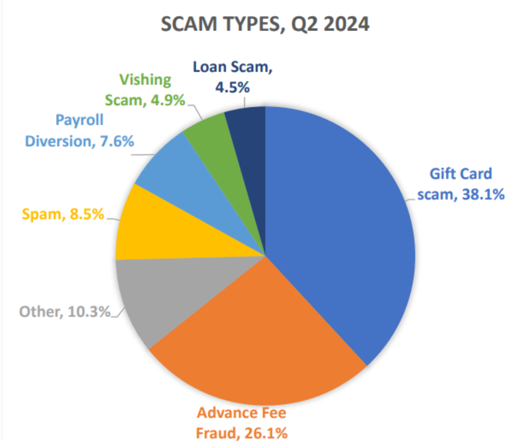

Fig. 2: Statistics of methods used in phishing attacks. [9]

According to Statista’s research, approximately 6 million unique phishing sites were detected between the first quarters of 2023 and 2024. [15] This is a pretty high number considering there are many ways to do a phishing attack. So keep in mind that this type of incident can happen to almost anyone. These steps will greatly help protect you from phishing attacks.

The main focuses of the study are:

- To address the methods, progress, results and protection methods of phishing attacks.
- The URL structure of the site will be examined with machine learning algorithms and classified as phishing or legitimate.
- With the database obtained from UCI, 134,850 legitimate and 100,945 phishing sites will be examined along with many features (SSL certificate availability, domain status etc.).
- RF, SVM Naive Bayes and C4.5 will be used from machine learning algorithms.
- Python will be used for the use of the algorithms and R language will be used for the graphical representation of the data obtained as a result of the study.

II. Literature Review

Various studies have been conducted on the detection of phishing sites with the help of artificial intelligence. Kang Leng Chiew and his team tried many machine learning methods using the Hybrid Ensemble Feature Selection (HEFS) method with the training set they received from UCI in 2019; They detected phishing emails with an accuracy rate of %94.6 with Support Vector Machines (SVM), Naive Bayes, C4.5 and Random Forest (RF) algorithms. [16] In the study conducted by Adem Korkmaz and Selma Büyükgöze in 2019, they used similar machine learning methods with the data set they received from UCI. [7] In this data set, there are 1353 websites, 548 legitimate, 702 phishing and 103 suspicious websites. Among the machine learning methods (Random Forest, SVM, Naive Bayes, C4.5), the most successful algorithm was Random Forest with % 95.3. In their study in 2024, Fazle Rabbi and his team used machine learning algorithms; Random Forest, Gradient Boosting, AdaBoost, Logistic Regression, KNN and Naive Bayes to detect phishing emails using Ling Spam and TREC datasets. [17] In the study, the Random Forest algorithm, in particular, showed the best performance with % 98.42 accuracy on the TREC dataset. In addition, it was found that the inclusion of the email subject line increased the accuracy of the model. In another study, Sindhu and his team, in their 2020 article, examined the detection of phishing websites using algorithms such as Random Forest (RF), Support Vector Machines (SVM), and Artificial Neural Networks (ANN) trained with Backpropagation. [18] In this study, it was determined that the Random Forest algorithm in particular gave the highest performance with an accuracy rate of 94.6%. In the study they published in 2018, Minh Nguyen and his team modeled email content at both the word and sentence level using Hierarchical Long Short-Term Memory Networks (H-LSTM) and Supervised Attention mechanisms.[19] With this approach, they achieved an accuracy rate of 97.8% in the detection of phishing emails. The study determined that deep learning techniques provide an effective solution in the field of cybersecurity. In the study conducted by Almujahid and his friends in 2024, they applied machine and deep learning techniques in the detection of phishing sites. As a result of this study, they achieved the most successful rate of 99% by using the CNN model.[20] In the study conducted by Almousa and his colleagues in 2022, they used deep learning algorithms to detect phishing sites. [21] They applied LSTM and CNN models on four different datasets, using the necessary features such as URL analysis. As a result of this study, it was determined that the LSTM model was the most successful model with 97.37%. In their study in 2024, Albishri and his friends worked with a dataset consisting of 1,561,932 URLs using NB, DT, RF, Gradient Boosting and other machine learning algorithms to detect phishing URLs. [22] In their analysis with 12,000 randomly selected samples from this dataset, they achieved 99.96% accuracy and 99.98% F1 score with DT. Olaolu Kayode-Ajala used machine learning algorithms to detect phishing sites in his study in 2022. [23] SVM, KNN, DT and RF algorithms were tested on a dataset consisting of 6,157 legitimate and 4,898 phishing URLs. RF was the most successful algorithm with 97%. In the study conducted by Taofeek Olayinka Agboola in

2024, in addition to machine learning algorithms such as XGBoost, RF, MLP, SVM, a TensorFlow-based method was preferred for visual classification. [24] In this study, although an accuracy rate of % 94.16 was achieved, it was mentioned that a higher rate could be achieved in the future by using GPU for CNN classification. In the study conducted by Karim et al. in 2023, a similar URL-based detection method was followed. [25] In the study, various machine learning models (RF, NB, SVM) and their hybrid combination, the LSD model, were used to classify phishing URLs. The LSD model exhibited the best performance with an accuracy rate of 98.12%.

In a study conducted by Aljofey and his colleagues, they tried to detect phishing attacks using CNN on URLs. [28] In this study, an accuracy rate of 95.22% and an F1 score of 95.16% were obtained. Ramanathan and his colleagues developed a machine learning method called ”phishGILLNET” in their study in 2012. [27] phishGILLNET is a three-stage phishing email detection method. Each stage aims to fill in the gaps of the previous stage. In the first stage, phishGILLNET1, e-mails were separated into two separate classes as ”phish” and ”non-phish” by using Probabilistic Latent Semantic Analysis (PLSA) to analyze the language (spelling mistakes, etc.). In phishGILLNET2, they combined the data they received from PLSA with AdaBoost to fill in the gaps in the first stage and to reach a higher accuracy rate. In the study conducted with approximately 400,000 e-mails, phishGILLNET2 achieved an accuracy rate of 99.7%.[27] In the study conducted by Tamal and his friends, 41 features were examined on 274,446 URLs with machine learning methods, and it was determined that the most successful method was RF with 97.5%. [29] In the study conducted by Abdul Samad and his friends, UCI and Mendeley data sets were used. [30] In this study where 21,055 samples were used, it was determined that GB and RF algorithms reached the highest accuracy. In the study conducted by Koçyiğit and his friends in 2024, they used Genetic Algorithms (GA) differently from other studies.[31] Since this algorithm provides optimization, better values are obtained. In RF, 92.93% accuracy value was reached. In the research conducted by A.K. Jain and B.B. Gupta, features in URL and source codes were examined with machine learning algorithms.[32] They worked with 4,059 samples obtained from Alexa, PhishTank and OpenFish. As a result, RF reached a verification rate of %99.09. In the study conducted by Shahrivari et al., 11,000 samples were examined with models such as Logistic Regression, DT, RF, XGBoost and Neural Networks. [33] As a result, the most successful method was %98.3 with XGBoost. In the study conducted by Musa et al. in 2020, they aimed to use fewer features in phising detection and reach higher accuracy values thanks to XGBoost. [34] The dataset they received from UCI contains 2456 samples and 30 features. While the accuracy rate of 97.29% was achieved with full features, the accuracy rate was increased to 97.41% by selecting only important features. These results showed that the algorithm is able to achieve high accuracy with low cost.

As seen from this table and the studies mentioned, the Random Forest (RF) algorithm is extremely effective in detecting phishing sites.

III. Method

A. Dataset

1) Data Source: The dataset named ”PhiUSIIL Phishing URL (Website)” published by UCI Repository was used.[35]

2) Data Type: In order to express the features in this data set, three different data types were used: Integer, Categorical and Continuous.

3) Data Size: Of the 235,795 URLs, 134,850 belong to legitimate websites and 100,945 belong to phishing websites. The dataset includes 56 different features of the websites (e.g. SSL certificate status, domain age, URL length, etc.).

B. Pre-processing

The dataset was checked for any inconsistencies or duplicate records. As a result, no inconsistencies or duplicate/missing records were detected. In addition, it was determined that the dataset was balanced (%57 Phising, %42 Legitimate) and class balancing methods were not needed.

C. Feature Extraction

Feature extraction is the process of obtaining meaningful and distinctive features from raw data and directly affects the performance of classification algorithms. [36] In this study, 56 features of the dataset were examined to distinguish phishing sites. Features that could be detected by machine learning algorithms were kept, and those that were unlikely to be useful were removed. In addition to the features of the dataset, two new features that could be useful were added. Each feature was separated by parameters to distinguish between legitimate and phishing sites:

- **URL Length**: URL length was taken directly from the dataset. The length value was categorized as short and long. The threshold value was selected as 35.
- **Domain Length**: The character length of the domain name was calculated and classified. The threshold value was selected as 20.
- **Number of Subdomains (NoOfSubDomain)**: Considering that the number of subdomains in URLs is higher, the number of subdomains was examined and classified as a threshold value of 2.
- **HTTPS Usage (IsHTTPS)**: The importance of implementing the HTTPS protocol was mentioned in the introduction. If it is used, it is classified as legitimate.
- **URL Similarity Index (URL SimilarityIndex)**: The similarity index was used because URLs with a high similarity rate to a legitimate site are more likely to be phishing. The threshold value was chosen as 0.8.
- **TLD Validity Probability (TLDLegitimateProb)**: The top-level domain (TLD) level of the domain name was chosen as a criterion. Domain names such as ”.com, .org, .net” have high reliability, having one of these domain names can make the examined site legitimate. The threshold value was chosen as 0.3.
- **ObfuscationRatio**: This ratio, which measures the complexity of the URL, was used as a determinant because it was high in phishing URLs. The threshold value was chosen as 0.1.
- **HasDescription**: The presence of a description field can put the site in the legitimate category. In this study, if the site does not have a description field, it is classified as phising.
- **External Form Submission (HasExternalFormSubmit)**: Whether forms are directed to external sources can be helpful in determining the status of the site. If data is transferred to external sources, it falls into the phising category.
- **Domain Name and Title Match (DomainTitleMatchScore)**: Using the domain name and title match score, the site falls into the phising category in cases where there are no similarities.

Article Authors	Dataset	Sample Numbers	Methods	Results	Deficiencies	Future Contributions
Fazle Rabbi et al. [17]	Ling Dataset, TREC Dataset	65.701 Sample (Ling: 2.876 TREC: 62.825)	NLP, RF, AdaBoost, KNN	Accuracy:%98,38 Sensibility: %98 F1:%98,38	Only the ”subject” and ”message” attributes are used. Sender information and URLs are ignored.	Incorporating other features into the model.
Chiew et al. [16]	UCI Repository	11.000 Sample	RF, SVM, Naive Bayes, C4.5, HEFS	Accuracy:%94,6	There is a lack of information about the class balance of the dataset used.	HEFS can be integrated with deep learning algorithms.
Korkmaz & Büyükgöze [7]	UCI Repository	1.353 Sample (Legit: 548, Phising: 702, Suspicious: 103)	DT, RF, SVM	Accuracy:%95,6 Sensibility:%97,2 F1:%95,3	The number of samples in the dataset is small and a single dataset is used.	A larger data set may be preferred.
Almujahid et al. [20]	Mendeley & UCI Repository	21.055 Sample	CNN, KNN, SVM, RF, XGBoost	Accuracy:%99 Specificity:%98 Sensibility:%98	No significant deficiency was detected.	More datasets and algorithms can be included.
Almousa et al. [21]	Tan, Kumar, UCI, AZA	72.605 Sample	LSTM, CNN, RF, C4.5	Accuracy:%97,37 Specificity:%90 Sensibility:%70	The success rate of the models decreased depending on the data set.	DMore data sets and models can be worked with.
Albishri et al. [22]	Public Sources	1.561.932 Sample (9.600 eğitim, 2.400 test)	NB, DT, RF, SVM, KNN	Accuracy:%99,96 F1: %99,98	Ineffective against zero-day attacks.	Deep learning algorithms can be used.
Olaolu Kayode-Ajala [23]	Public Sources	11.055 Sample (Legit: 6,157, Phishing: 4,898)	DT, KNN, SVM, RF	Accuracy:%97 F1: %93	The source of the dataset is not clearly stated.	More detailed information about the materials used can be provided.
Orunsolu et al. [26]	PhishTank	5041 Sample (Legit: 2500, Phising: 2541)	SVM, NB	Accuracy:%99.96	It was studied on a small dataset.	It is possible to work on a larger data set.
Taofeek Olayinka Agboola [24]	PhishTank	10.000 Sample and 250 Pictures	DT, RF, XG-Boost, SVM	Accuracy:%94,16	Limited dataset, false positive and negative rates, and low-performance GPU usage.	Better results can be achieved by using a more powerful GPU.
V. Ramanthan et al. [27]	SpamAssassin, PhishingCorpus, Enron Email Dataset, SPAM Archive, PhishTank	400.000 Sample	PLSA, AdaBoost, NLP	Accuracy:%97,7 Sensibility:%97,7 F1: %97,7	The study was conducted on English texts only.	The number of languages examined may be increased.
Aljofey et al. [28]	Alexa, OpenFish, Phishtank, Spamhaus.org	318.642 Sample (Legit: 157.626 Phishing: 161.016)	CNN	Accuracy:%95,22 Sensibility:%95,01 F1: % 95,16	The training period is too long.	The model architecture can be further optimized.
Abdul Karim et al. [25]	Kaggle	11.054 Sample	DT, LR, RF, NB, KNN, SVM	Accuracy:%96,77 Specificity:%95,83 Sensibility:%96,73 F1: %97,12	A small dataset was used and some algorithms showed poor performance.	A larger data set should be used.

TABLE I: Literature Review Results

Article Authors	Dataset	Sample Numbers	Methods	Results	Deficiencies	Future Contributions
Tamal et al. [29]	OpenPhish, DomCop, Aalto University	274.446 Sample (Legit: 139.946, Phising: 134.500)	RF, ExtraTreesClassifier	Doğruluk:%97,5 Hassasiyet:%97 F1: %98	Deep learning algorithms could not be used due to hardware limitations.	Deep learning algorithms can be used.
Abdul Samad et al. [30]	UCI ve Mendeley	21.055 Sample (Legit: 11.157, Phishing: 9.898	SVM, RF, GB, DT, KNN	Accuracy:%98.26 Sensibility:%98,5 F1: %98,2	Not testing performance against new phishing types.	Increasing performance by adding new features.
Koçyiğit et al. [31]	Kaggle	87.489 Sample (Legit: 51.316, Phishing: 36.173)	GA, XGBoost, KNN, SVM, RF, NB	Accuracy:%92.93 Sensibility:%93.45	Only URL-based features were examined.	Features related to HTML and CSS can be examined.
A.K. Jain & B.B. Gupta [32]	Alexa, Phishtank, Openphish	4059 Sample (Legit: 1.918, Phishing: 2.141)	SVM, RF	Accuracy:%99,09	Transactions can only be made on HTML-based sites.	Classification can also be done on sites that are not based on HTML.
Shahrivari et al. [33]	Kaggle	11.000 Sample (Legit: 6.157, Phishing: 4.898)	AdaBoost, DT, SVM, RF, KNN, XGBoost	Accuracy:%98,3 Sensibility:%98,7 F1: %97,6	A single type of data set was used.	Hybrid methods can be used.
Musa et al. [34]	UCI Repository	2456 Sample	XGBoost	Accuracy:%97,29 MCC:%94,49 F1:%97,24	It is not focused on emails containing advertising etc.	The circle of content to be examined can be expanded.

TABLE II: Literature Review Results

- **Number of Redirects (NoOfURLRedirect)**: The number of times the URL is directed to other sites will be decisive in detecting phising sites. Excessive redirection increases the risk. In this study, the redirection threshold value was selected as 1.
- **Number Presence in Domain (HasNumberInDomain)**: The domain of a significant majority of websites starts with ”www”. Some sites’ domains contain numbers (e.g. www123). This increases the likelihood of the site being phished. Domains will be examined with Regex and the presence/absence of numbers will be detected.
- **Special Character Count (SpecialCharCount)**: Legitimate sites have few special characters in their URLs. If the number of special characters is high, the site is likely to be phishing. URLs will be examined with Regex and phishing sites will be detected.

D. Feature Selection

When selecting features, attention was paid to the meaningfulness of the features. Features such as URL length and number of subdomains were selected as a priority. After the features were determined, each was classified with certain parameters. The features and parameters used are given in the table below.

Feature	**Description and Parameters**
URL Length (URLLength)	URL Length >35 → Phising; URL Length ≤ 35 → Legit.
Domain Name Length (DomainLength)	Domain Name Length >20 → Phishing; Domain Name Length ≤ 20 → Legit.
Number of Subdomains (NoOfSubDomain)	Number of subdomains >2 → Phishing; Number of subdomains ≤ 2 → Legit.
HTTPS Presence (IsHTTPS)	If HTTPS is used → Legit; If not → Phising.
URL Similarity Index (URLSimilarityIndex)	Similarity score >0.8 → Phishing; Similarity score ≤ 0.8 → Legit.
TLD Legitimate Probability (TLDLegitimateProb)	Legitimate probability < 0.3 → Phishing; Legitimate probability ≥ 0.3 → Legit.
Obfuscation Ratio (ObfuscationRatio)	Obfuscation ratio >0.1 → Phishing; Obfuscation ratio ≤ 0.1 → Legit.
Is There a Description Field? (HasDescription)	If there is → Legit; If not → Phishing.
External Form Submission (HasExternalFormSubmit)	If it directs → Phising; If not → Legit.
Domain Name and Title Compatibility (DomainTitleMatchScore)	Compliance score < 0.5 → Phishing; Compliance score ≥ 0.5 → Legit.
Number of Redirects (NoOfURLRedirect)	Redirects >1 → Phishing; Redirects ≤ 1 → Legit.
Number Existence in Domain (HasNumberInDomain)	If there is → Phishing; If not → Legit.
Number of Special Characters (SpecialCharCount)	Characters >5 → Phishing; Characters ≤ 5 → Legit.

TABLE III: Features and parameters used to detect phishing sites.

E. Classification

Machine learning is an artificial intelligence discipline that uses data to perform a task and works through decision making. [37] In this study, four different machine learning algorithms were used to perform classification.

1) Random Forest (RF): Random Forest is an ensemble learning method that works by creating multiple decision trees. This algorithm allows each tree to make independent predictions and then reaches the conclusion with a majority vote. In this way, the error rate of the model is reduced and it exhibits strong performance, especially on large data sets. Considering the large data set we will use in this study and the success of Random Forest in similar studies, its importance in the study is great.[38]

2) Support Vector Machines (SVM): Support Vector Machines are an effective classification algorithm for two-class problems. In cases where the data is not linearly separated, the data is transformed into a higher-dimensional space using kernel functions. With this transformation, non-linear boundaries can also be created and more complex classifications can be made. [39]

3) Naive Bayes: Naive Bayes is a probability-based classification algorithm. It classifies data using Bayes Theorem. Thanks to the assumption of independence between features, the algorithm works very fast and is generally preferred in high-dimensional data sets such as text classification. Since a large data set will be used in this study, its speed is an advantage. [40]

4) C4.5 Decision Tree: C4.5 is an advanced version of decision tree algorithms. This algorithm uses an entropy-based approach to reduce uncertainty in the data set. C4.5 can work with both numerical and continuous data and thus can make effective classifications in a wide range of data. Considering that the features selected in this study are both numerical and continuous, the C4.5 algorithm is of great importance. [38]

5) AdaBoost: AdaBoost is known as an adaptive boosting algorithm and its main purpose is to combine multiple weak learners to create a strong classifier. The main reasons for using AdaBoost in this study are that the algorithm performs well especially on small and medium-sized datasets and its success in reducing error rates. Past studies have shown that AdaBoost achieves high accuracy rates in detecting phishing sites and generally performs fast calculations.[34]

IV. Experimental Results

A. Model Performance and Evaluation

1) Performance Metrics: Performance metrics are used to evaluate the success of machine learning and deep learning models. Metrics provide information about performance by analyzing different aspects of models.

Accuracy is the easiest and most valid method used to measure model performance. The calculation of the accuracy value is provided by True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN) values. The ratio of the number of incorrectly classified examples (FP+FN) to the total number of examples (TP+TN+FP+FN) gives us the accuracy value.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision measures the ratio of the model's true positive predictions to all positive predictions. For example, if the precision value is high for a spam filter model, the vast majority of emails that the model marks as "spam" will actually be spam.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Sensitivity measures how accurately the model detects true positives. In this study, it will show how many of the real phishing sites the model correctly classifies as "phishing."

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

F1 Score provides a balance between precision and sensitivity and is calculated as the harmonic mean of both metrics for its value. It ensures that the model keeps both false positives (False Positives) and false negatives (False Negatives) to a minimum. Considering its importance in this study, high Recall alone will not be enough to detect phishing sites, because a high number of false positives (for example, a secure site being incorrectly labeled as phishing) means pushing the user to the phising site. F1 score will increase the detection of phishing sites and prevent the false marking of secure sites by balancing precision and sensitivity.

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}}$$

AUC is the area under the ROC (Receiver Operating Characteristic) curve of a model. This metric measures the overall classification performance of the model. In imbalanced datasets such as phishing detection (for example, in many datasets, although not in this study, the number of phishing sites is much lower than the number of legitimate sites), AUC is an ideal metric to evaluate the overall performance of the model. In particular, it shows the ability to distinguish phishing and safe sites at different threshold values. An AUC value of 1 can be classified as excellent, greater than 0.8 as strong, and less than 0.8 as weak.

Principal Component Analysis (PCA) is the process of calculating the principal components and using them to perform a fundamental change on the data. Datasets used in the detection of phishing sites usually contain a large number of features. These features consist of various factors such as the structural analysis of the URL, domain name information, and the presence of HTTPS. However, some features may show high interaction with each other or may contain unnecessary information that does not contribute to the performance of the model. When PCA is not used, i.e. when it is OFF, all features are given to the model as is. However, since all the data will be used in this case, it will slow down the model training process considerably. Therefore, it is healthier to use PCA in large datasets. In this study, both cases where PCA is ON and OFF are examined and compared.

Model	**Accuracy**	**Senstivity**	**Precision**	**F1**	**AUC**
RF	0.9997667	0.9997667	0.9997668	0.9997667	0.9999997
SVM	0.9993002	0.9993002	0.9993007	0.9993002	0.9999966
NB	0.9312750	0.9312750	0.9383701	0.9302474	0.9751056
C4.5	0.9989186	0.9989186	0.9989185	0.9989185	0.9988864
AdaBoost	0.9992366	0.9992366	0.9992366	0.9992366	0.9999977

TABLE IV: Performance metrics of machine learning models in PCA OFF case.

Model	**Accuracy**	**Senstivity**	**Precision**	**F1**	**AUC**
RF	0.9968405	0.9968405	0.9968406	0.9968402	0.9999200
SVM	0.9992260	0.9992260	0.9992261	0.9992260	0.9999775
NB	0.9621917	0.9621917	0.9623865	0.9622319	0.9865835
C4.5	0.9940838	0.9940838	0.9940850	0.9940842	0.9940243
AdaBoost	0.9922602	0.9922602	0.9922607	0.9922604	0.9996963

TABLE V: Performance metrics of machine learning models in the PCA ON case.

2) PCA Effect on Model Performance: In the case of PCA ON, there was a slight decrease in metrics such as accuracy and sensitivity. However, these differences are so small that they can be ignored. The largest impact was seen in the Random Forest and SVM models.

3) Error Analysis: The error analysis phase is of critical importance in machine learning and similar projects. This phase reveals the strengths and weaknesses of the project and provides an understanding of whether the work has a real-life counterpart. If the project is to be developed in the future, the results obtained in the error analysis are analyzed and more appropriate steps are selected for development. Accordingly, while evaluating the performance of the models used in the project, confusion matrices were prepared to examine the correct and incorrect classification results of each model in detail. Confusion matrices show the True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) values of the models, allowing us to understand the error types more clearly. The confusion matrices and performance metrics of the models used are given in the table below.

Model	TP	FP	FN	TN	Accuracy(%)
RF	20180	9	2	26968	99.97
SVM	20160	29	4	26966	99.92
NB	16975	3214	27	26943	95.58
C4.5	20162	27	24	26946	99.89
AdaBoost	20167	22	14	26956	99.93

TABLE VI: The table shows the accuracy percentage (%) of each model, as well as the true positive (TP), false positive (FP), false negative (FN), and true negative (TN) values.

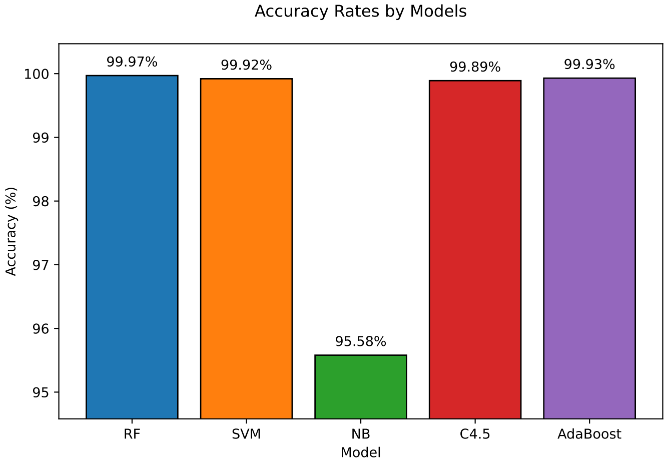

Fig. 3: Accuracy rate of models

According to the results here, the highest accuracy rate was achieved with Random Forest. It is also seen that it can distinguish classes well in terms of both FP and FN. Although Naive Bayes has a high accuracy rate, it is not suitable for this application due to the FP and FN rates.

4) Training and Testing Period: Training and testing processes put a heavy load on resources such as the processor (CPU) or graphics processor (GPU). The longer the duration, the negative situation in terms of energy consumption and cost. Therefore, the shorter the total duration, the more efficient the system works. The table below shows the total training and testing times of the models.

5) Generalization Ability of the Model: RF, one of the models used in the study, is a model with strong generalization ability. Because it consists of a combination of random trees, it reduces the variance in the data. According to the results of this study, RF provided high accuracy on both training and test data and proved its generalization ability with low error rates (False Positive and False Negative values). Although SVM is also a model with high generalization ability, it can obtain unsuccessful results because it uses more resources in large data sets. Naive Bayes, on the other hand, is a limited model in terms of generalization. When the results in the confusion matrix table are examined, it is clearly seen that it lags behind other models in terms of accuracy rate.

6) Cross Validation Results: Cross-validation evaluates the generalization ability of the model by measuring how the model performs on different subsets of data. The cross-validation results obtained in this study are given.

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
RF	0.999655	0.999681	0.999814	0.999681	0.999681
SVM	0.999204	0.999337	0.999416	0.999549	0.999204
NB	0.932569	0.933654	0.932780	0.933654	0.932568
C4.5	0.999019	0.998886	0.999019	0.999072	0.998992
AdaBoost	0.999363	0.998860	0.999257	0.999337	0.999284

The scores obtained by the models in each fold are as follows. It will be easier to examine when the results of each model are averaged. The average cross-validation values of each model are shown in the graph below.

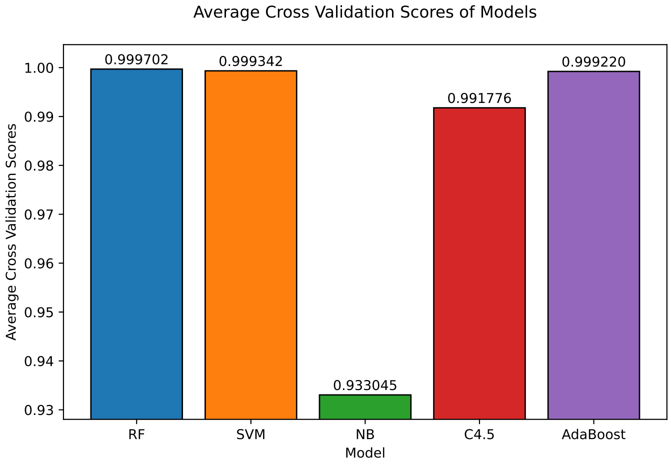

Fig. 4: Average cross-validation scores of the models

V. Discussion

A. *Algorithm Performances*

In this study, the performances of machine learning algorithms (RF, SVM, NB, C4.5 and AdaBoost) used for phishing detection were evaluated both individually and comparatively. The findings of the study show that the machine learning methods used generally achieved high success levels, despite minor differences in the characteristics of the dataset and the parameters determined in the study. The most successful method with 99.7% accuracy rate was Random Forest, while the least successful method with 93.1% was Naive Bayes.

B. *Dataset and Effect of Features*

The study examined 235,795 URLs. Of these, 134,850 belong to legitimate websites and 100,945 belong to phishing websites. When examined in percentages, 57% is phishing and 43% is legitimate, which reveals that the data set is balanced.

C. *Time and Source Usage*

In this study, a very large data set was used and 70% of the samples were examined as training and 30% as test. When the models were compared in terms of the time it took to complete all operations, it was determined that the fastest model was Naive Bayes with 7.75 seconds. The slowest model among them was

SVM with 771.33 seconds. When we look at the effect of these times on the accuracy rate, although Naive Bayes was the fastest model, it achieved the lowest accuracy rate. Despite the high accuracy of SVM, the fact that its processing time is extremely long compared to other models can make it difficult to use in large-scale systems. The most successful model, Random Forest, completed all operations in 258.32 seconds. With its high performance and average computation time, the most efficient model after Random Forest was AdaBoost. AdaBoost completed all operations in 204.24 seconds. The Randon Forest and AdaBoost algorithms did a successful job with both high accuracy and reasonable processing times. The success of Random Forest is clearly seen in similar studies in the literature. No information is provided regarding the resource usage of the models in the study.

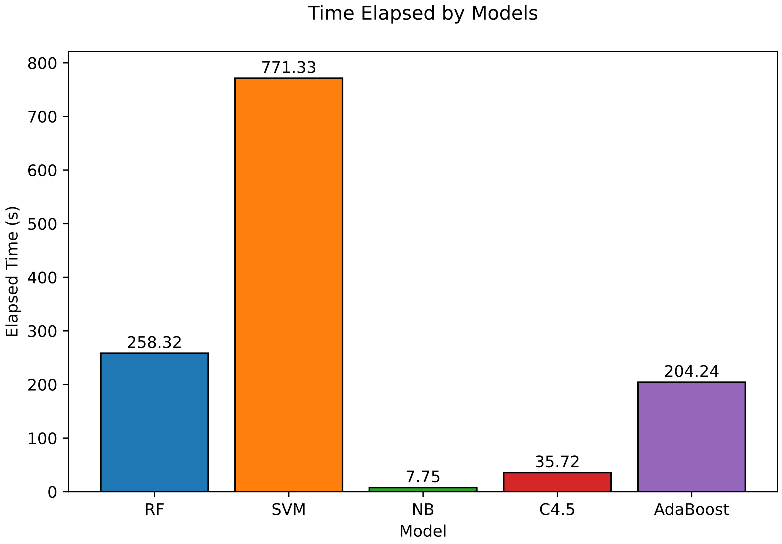

Fig. 5: Time taken for calculations based on models.

D. Original Contributions

In the study, two features were added that were not directly included in the data set but could provide results when analyzed on URLs and were thought to contribute to the study. The ”Number of Special Characters” and ”Number Presence in the Domain” features were included in the study within the specified parameters and increased the success rate.

E. Limitations and Future Contributions

The accuracy rates obtained in the study are very high despite the size of the data set and the abundance of features used. This situation requires three different queries. The first situation; there may have been over-learning, when this situation occurs, high accuracy rates are obtained, but it is eliminated when the cross-validation results without over-learning are also high. The second situation; the study was indeed successful and it can be thought that all models obtained high results, but high results do not always indicate that the study was ”successful”. The third and last situation is that an error may have been made in the classification and a too easy path was drawn for the examination of the models. This is the biggest problem that can be mentioned as a limitation in the study. The models; make two different classifications as ”legitimate” and ”phishing” according to the features and parameters. If such a sharp classification had not been made in the classification and an ”intermediate class” had been created (for example, legitimate - suspicious - phishing), the accuracy rates would probably have been lower, but more realistic results would have been obtained and the models would have passed a real test.

Only machine learning methods were applied in the study. Deep learning methods (such as CNN and LSTM) can be integrated in the future, and if the data set does not offer features suitable for deep learning methods (for example, the current data set is not suitable for image processing), data sets obtained from different sources can be included and examined in the study.

VI. Conclusion

In this study, it is shown that fake and phishing websites can be classified with URL-based features using machine learning algorithms. Random Forest, Support Vector Machines, Naive Bayes, C4.5 and AdaBoost algorithms were used from machine learning algorithms. Random Forest algorithm exhibited the highest performance with 99.97% accuracy rate and it was observed that the error rate was low compared to other methods.

References

[1] Datareportal. Digital 2024: Deep dive - the state of internet adoption, 2024. Erişim: 2024-11-13.

[2] Jason Hong. The state of phishing attacks. *Commun. ACM*, 55(1):74–81, January 2012.

[3] Ümit Sönmez. Bilişim sistemleri araciliğiyla dolandiricilik suçu. *Dicle Üniversitesi Adalet Meslek Yüksekokulu Dicle Adalet Dergisi*, 1(2):47–68, 2017.

[4] Remzi GÜRFİDAN. Intelligent methods in cyber defence: Machine learning based phishing attack detection on web pages. *Mühendislik Bilimleri ve Tasarım Dergisi*, 12(2), 2024.

[5] Zainab Alkhalil, Chaminda Hewage, Liqaa Nawaf, and Imtiaz Khan. Phishing attacks: A recent comprehensive study and a new anatomy. *Frontiers in Computer Science*, 3:563060, 2021.

[6] Nagaraju Pureti. Phishing scams: How to recognize and avoid becoming a victim. *International Journal of Machine Learning Research in Cybersecurity and Artificial Intelligence*, 15(1):51–73, 2024.

[7] Adem Korkmaz and Selma Büyükgöze. Sahte web sitelerinin sınıflandırma algoritmaları ile tespit edilmesi. *Avrupa Bilim ve Teknoloji Dergisi*, (16):826–833, 2019.

[8] Sultan Asiri, Yang Xiao, Saleh Alzahrani, Shuhui Li, and Tieshan Li. A survey of intelligent detection designs of html url phishing attacks. *IEEE Access*, 11:6421–6443, 2023.

[9] Phishing activity trends report - 2nd quarter 2024, 2024. Accessed: 2024-11-14.

[10] Cloudflare. What is business email compromise ?

[11] Eiman Al Neyadi, Shaima Al Shehhi, Ameera Al Shehhi, Noora Al Hashimi, Qbea’H Mohammad, and Saed Alrabaee. Discovering public wi-fi vulnerabilities using raspberry pi and kali linux. In *2020 12th Annual Undergraduate Research Conference on Applied Computing (URC)*, pages 1–4. IEEE, 2020.

[12] Yusuf Alaca. Siber güvenlikte cic-darknet2020 veri seti kullanarak vpn/novpn ve tor/notor sınıflandırması: Basit ve karmaşık modellerin kullanımı. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 35(2):569–579, 2023.

[13] George E Violettas and Kyriakos Papadopoulos. Passwords to absolutely avoid. In *The Fifth International Conference on*

the Applications of Digital Information and Web Technologies (ICADIWT 2014), pages 60–68. IEEE, 2014.

[14] İlhan Fırat Kılınçer, Fatih Ertam, Orhan Yaman, and Abdülkadir Şengür. An effective security method based on combining 802.1 x, dmz and ssl-vpn for iot network security. *Acta Infologica*, 4(2):65–76, 2020.

[15] Statista Research Department. Number of phishing domain names worldwide from 2013 to 2023, 2024. Accessed: 2024-11-14.

[16] Kang Leng Chiew, Colin Choon Lin Tan, Koksheik Wong, Kelvin Yong, and Wei Tiong. A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. *Information Sciences*, 484:153–166, 05 2019.

[17] Md Fazle Rabbi, Arifa I Champa, and Minhaz F Zibran. Phishy? detecting phishing emails using machine learning and natural language processing. In *Software Engineering and Management: Theory and Application: Volume 16*, pages 119–137. Springer, 2024.

[18] Smita Sindhu, Sunil Parameshwar Patil, Arya Sreevalsan, Faiz Rahman, and Ms Saritha AN. Phishing detection using random forest, svm and neural network with backpropagation. In *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, pages 391–394. IEEE, 2020.

[19] Luong Anh Tuan Nguyen, Ba Lam To, Huu Khuong Nguyen, and Minh Hoang Nguyen. A novel approach for phishing detection using url-based heuristic. In *2014 international conference on computing, management and telecommunications (ComManTel)*, pages 298–303. IEEE, 2014.

[20] Noura Fahad Almujahid, Mohd Anul Haq, and Mohammed Alshehri. Comparative evaluation of machine learning algorithms for phishing site detection. *PeerJ Computer Science*, 10:e2131, 2024.

[21] May Almousa, Tianyang Zhang, Abdolhossein Sarrafzadeh, and Mohd Anwar. Phishing website detection: How effective are deep learning-based models and hyperparameter optimization? *Security and Privacy*, 5(6):e256, 2022.

[22] Adel Ataih Albishri and Mohamed M Dessouky. A comparative analysis of machine learning techniques for url phishing detection. *Engineering, Technology & Applied Science Research*, 14(6):18495–18501, 2024.

[23] Olaolu Kayode-Ajala. Applying machine learning algorithms for detecting phishing websites: Applications of svm, knn, decision trees, and random forests. *International Journal of Information and Cybersecurity*, 6(1):43–61, 2022.

[24] Agboola Olayinka Taofeek. Development of a novel approach to phishing detection using machine learning. *ATBU Journal of Science, Technology and Education*, 12(2):336–351, 2024.

[25] Abdul Karim, Mobeen Shahroz, Khabib Mustofa, Samir Brahim Belhaouari, and S Ramana Kumar Joga. Phishing detection system through hybrid machine learning based on url. *IEEE Access*, 11:36805–36822, 2023.

[26] Abdul A Orunsolu, Adesina S Sodiya, and AT Akinwale. A predictive model for phishing detection. *Journal of King Saud University-Computer and Information Sciences*, 34(2):232–247, 2022.

[27] Venkatesh Ramanathan and Harry Wechsler. phishgillnet—phishing detection methodology using probabilistic latent semantic analysis, adaboost, and co-training. *EURASIP Journal on Information Security*, 2012:1–22, 2012.

[28] Ali Aljofey, Qingshan Jiang, Qiang Qu, Mingqing Huang, and Jean-Pierre Niyigena. An effective phishing detection model based on character level convolutional neural network from url. *Electronics*, 9(9):1514, 2020.

[29] Maruf A Tamal, Md K Islam, Touhid Bhuiyan, Abdus Sattar, and Nayem Uddin Prince. Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning. *Frontiers in Computer Science*, 6:1428013, 2024.

[30] Saleem Raja Abdul Samad, Sundarvadivazhagan Balasubaramanian, Amna Salim Al-Kaabi, Bhisham Sharma, Subrata Chowdhury, Abolfazl Mehbodniya, Julian L Webber, and Ali Bostani. Analysis of the performance impact of fine-tuned machine learning model for phishing url detection. *Electronics*, 12(7):1642, 2023.

[31] Emre Kocyigit, Mehmet Korkmaz, Ozgur Koray Sahingoz, and Banu Diri. Enhanced feature selection using genetic algorithm for machine-learning-based phishing url detection. *Applied Sciences*, 14(14):6081, 2024.

[32] Ankit Kumar Jain and Brij B Gupta. Towards detection of phishing websites on client-side using machine learning based approach. *Telecommunication Systems*, 68:687–700, 2018.

[33] Vahid Shahrivari, Mohammad Mahdi Darabi, and Mohammad Izadi. Phishing detection using machine learning techniques. *arXiv preprint arXiv:2009.11116*, 2020.

[34] Hajara Musa, AY Gital, Mohzo Gideon Bitrus, NF Juma, and Muhammad Abubakar Balde. Boosting the accuracy of phishing detection with less features using xgboost. *International Journal of Software & Hardware Research in Engineering*, 8(2), 2020.

[35] Irvine University of California. PhiUSIIL Phishing URL Dataset. https://archive.ics.uci.edu/dataset/967/phiusiil+phishing+url+dataset, n.d. Accessed: 2024-11-29.

[36] Özge Akdoğan and Selma Ayşe Özel. Effects of feature extraction techniques on classification of turkish texts. *Çukurova Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, 34(3):95–108, 2019.

[37] Ting Li, Kasper Johansen, and Matthew F McCabe. A machine learning approach for identifying and delineating agricultural fields and their multi-temporal dynamics using three decades of landsat data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 186:83–101, 2022.

[38] Shiju Sathyadevan and Remya R. Nair. Comparative analysis of decision tree algorithms: Id3, c4.5 and random forest. In Lakhmi C. Jain, Himansu Sekhar Behera, Jyotsna Kumar Mandal, and Durga Prasad Mohapatra, editors, *Computational Intelligence in Data Mining - Volume 1*, pages 549–562, New Delhi, 2015. Springer India.

[39] Sonal Pathak, Suhail Javed Quraishi, Anupam Singh, Malikhan Singh, Kavita Arora, and Danish Ather. A comparative analysis of machine learning models: Svm, naïve bayes, random forest, and lstm in predictive analytics. In *2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS)*, pages 790–795, 2023.

[40] Ayşe Berna Altınel. Türkçe metinlerde makine öğrenmesi algoritmalarının duygu analizi problemi üzerindeki performansının kıyaslanması. *Avrupa Bilim ve Teknoloji Dergisi*, (28):1056–1061, 2021.