International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Phishing Website Identification Using Supervised Learning Models

Prof. Prabhu Kichadi[1]*, Shravana Janawad*[2]*, Shruti Abane*[3]*, Shivani Patil*[4]*, Rutuja Halijwale*[5]

[1] Asst. Professor, Dept of Computer Science & Engineering, VSM's Somashekhar R. Kothiwale Institute of Technology, Nipani, Karnataka, India, 591237.

[2] Dept of Computer Science & Engineering, VSM's Somashekhar R. Kothiwale Institute of Technology, Nipani, Karnataka, India, 591237

[3] Dept of Computer Science & Engineering, VSM's Somashekhar R. Kothiwale Institute of Technology, Nipani, Karnataka, India, 591237

[4] Dept of Computer Science & Engineering, VSM's Somashekhar R. Kothiwale Institute of Technology, Nipani, Karnataka, India, 591237

[5] Dept of Computer Science & Engineering, VSM's Somashekhar R. Kothiwale Institute of Technology, Nipani, Karnataka, India, 591237

ABSTRACT:

Phishing websites are a major threat to cybersecurity, as they imitate legitimate websites to trick users into sharing sensitive information like passwords and credit card details. This project presents a system to detect phishing websites using machine learning and deep learning techniques. Multiple models, including Decision Tree, Random Forest, Support Vector Machine (SVM), and Artificial Neural Networks (ANN), were trained and tested to find the most accurate and reliable method.

A dataset of over 11,000 website records was collected from PhishTank.org. Data preprocessing techniques such as normalization, feature selection, and dimensionality reduction using Principal Component Analysis (PCA) were applied to improve the performance of the models. Among all models tested, the ensemble learning approach provided the highest accuracy of 99%, making it the most effective in identifying phishing websites. While models like SVM had slightly lower accuracy, they still contributed valuable insights.

In addition, a simple and user-friendly interface was developed for real-time detection of phishing URLs. This project demonstrates the power of machine learning in improving cybersecurity and highlights how intelligent systems can help protect users from online fraud. The proposed solution can be further enhanced and integrated into browser extensions, email filters, or security software to provide real-time phishing protection.

Keywords:

Phishing Detection, Cybersecurity, Machine Learning, Deep Learning, Ensemble Learning, Decision Tree, Random Forest, Support Vector Machine (SVM), Artificial Neural Networks (ANN), Feature Selection, Principal Component Analysis (PCA), URL Classification, PhishTank Dataset, Real-Time Detection, Online Fraud Prevention, Intelligent Systems, Web Security, Data Preprocessing, Phishing Website Identification, User-Friendly Interface.

1. Introduction

Phishing has emerged as a highly adaptive and persistent form of cyberattack, posing significant risks to digital security. In a typical phishing scenario, attackers trick users into disclosing sensitive information—such as usernames, passwords, or financial credentials—by impersonating trustworthy sources. These attacks are executed through forged emails, fraudulent websites, or deceptive messages. As phishing methods become increasingly convincing, users face growing difficulty in identifying malicious intent, making conventional detection techniques less effective.

Advanced phishing methods like spear phishing and whaling have added a new layer of complexity. Spear phishing targets individuals with highly personalized content, often obtained through online data mining, while whaling is aimed at high-ranking individuals such as executives or government officials. These refined strategies make use of psychological manipulation and social engineering, effectively bypassing traditional, rule-based filters and highlighting the limitations of static security protocols.

A key tactic employed by attackers involves creating websites that closely imitate legitimate ones in terms of design, layout, and functionality. These counterfeit websites often replicate visual elements, such as company logos and page structures, and even utilize authentic-looking security indicators like HTTPS encryption and padlock icons. Such techniques exploit user trust and attention to detail, making it difficult even for vigilant users to distinguish fake sites from genuine ones. The Anti-Phishing Working Group (APWG) reported over five million phishing incidents in 2023, illustrating the growing scale and sophistication of these attacks.

Industries that frequently handle sensitive personal and financial information—such as online banking, e-commerce platforms, and social media networks—are particularly vulnerable to phishing threats. As these platforms rely heavily on user interaction and trust, attackers often exploit them to gather data and carry out fraudulent activities. Relying solely on user awareness and traditional blacklist-based systems is no longer sufficient in protecting these digital environments.

To counter modern phishing tactics, there is a growing need to employ intelligent and adaptive security systems. Machine Learning (ML), a branch of Artificial Intelligence (AI), has proven to be a robust solution for detecting phishing attempts. ML algorithms can process and analyze vast datasets to uncover hidden patterns and identify irregularities that are indicative of malicious behavior. Through supervised learning techniques, models are trained using labeled datasets that include both legitimate and phishing websites, enabling the system to accurately classify unseen URLs.

Important features used in phishing detection include URL characteristics, domain registration data, SSL certificate details, and page content metrics. By analyzing these attributes, machine learning models such as Decision Trees, Random Forests, Support Vector Machines (SVM), and Artificial Neural Networks (ANN) can make accurate predictions regarding the legitimacy of a website. Additionally, preprocessing techniques like normalization and Principal Component Analysis (PCA) enhance the efficiency and performance of these models by reducing dimensionality and improving feature relevance.

Current phishing detection strategies often combine multiple techniques, such as URL-based filtering, content inspection, visual similarity analysis, and machine learning classification. Incorporating ensemble methods and deep learning models further improves accuracy and adaptability, enabling systems to respond effectively to new and evolving attack patterns. These AI-based solutions provide not only scalability and high precision but also the ability to operate in real-time, making them indispensable tools in modern cybersecurity defenses.

Nomenclature

AI: Artificial Intelligence – A branch of computer science focused on building systems capable of performing tasks that normally require human intelligence.
ML: Machine Learning – A subfield of AI that involves training algorithms to identify patterns and make decisions based on data.
Phishing: A cyberattack method where attackers deceive users into revealing personal or sensitive information by impersonating legitimate sources.
Whaling: A phishing technique that targets high-profile individuals such as executives or government officials.
SSL Certificate: Secure Sockets Layer Certificate – A digital certificate that authenticates a website's identity and enables encrypted connections.
SVM: Support Vector Machine – A supervised learning model used for classification and regression tasks.
ANN: Artificial Neural Network – A computational model inspired by the human brain, used in deep learning.
PCA: Principal Component Analysis – A dimensionality reduction technique used to simplify datasets while retaining key features.
APWG: Anti-Phishing Working Group – An organization that collects and reports data related to phishing attacks worldwide.

1.1. Structure/ DFD

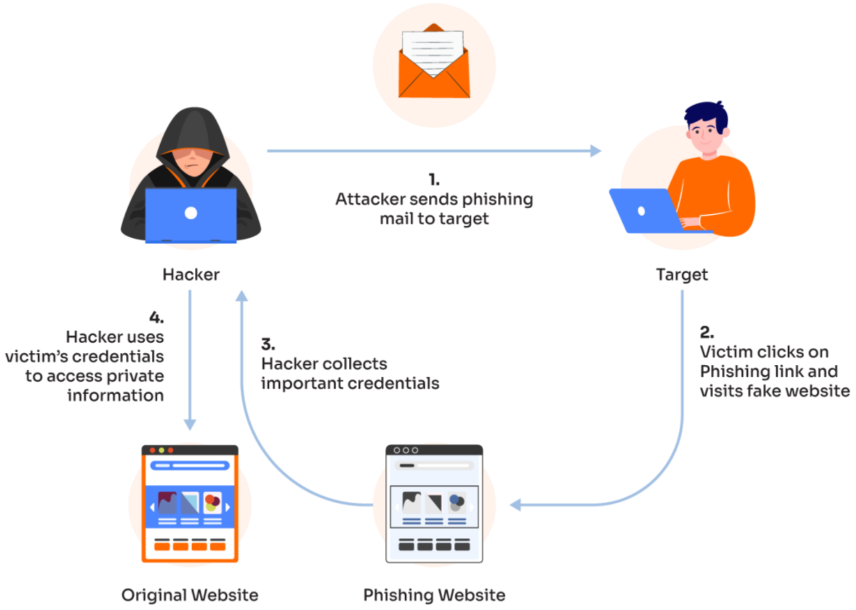
Phishing Attack Lifecycle: A Step-by-Step Breakdown

Phishing attacks generally follow a structured and deceptive sequence designed to manipulate users into revealing sensitive information. The diagram above illustrates the complete flow of a typical phishing attempt from initiation to exploitation. The process begins when a cyber attacker initiates contact by sending a fraudulent email to the target. This email is often disguised to appear as if it originated from a trusted organization, using familiar logos, language, and sender details to reduce suspicion. The embedded message typically urges the user to act swiftly—such as updating an account, verifying details, or responding to a security alert.

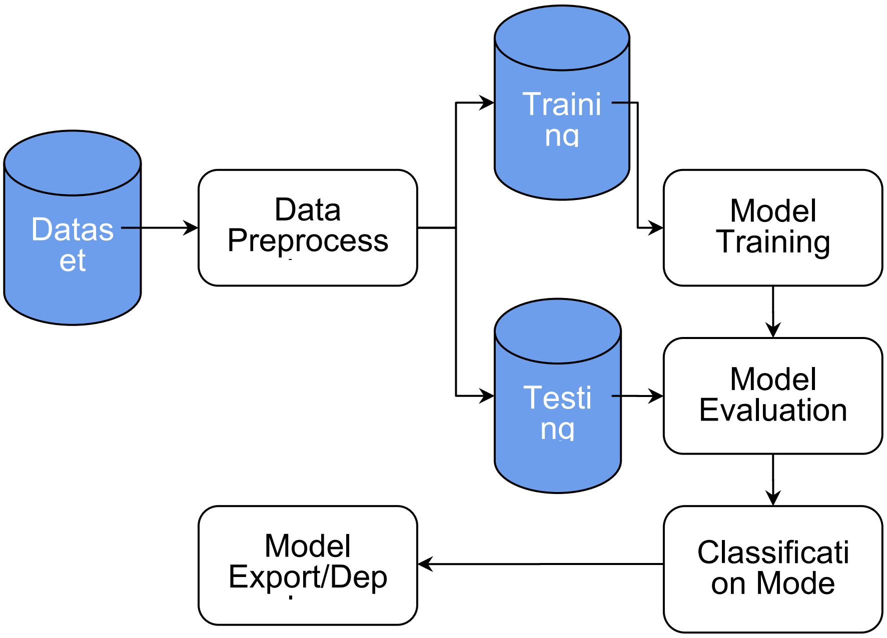
When the target user interacts with the email, usually by clicking on an embedded hyperlink, they are redirected to a counterfeit website. This phishing site is engineered to replicate the design, branding, and functionality of a legitimate website, often including visual elements such as secure-looking padlock icons and HTTPS in the URL. The aim is to convince the user that they are interacting with a legitimate service provider.

Unaware of the deception, the user proceeds to enter confidential data on the phishing website. This commonly includes login credentials, credit card details, or other personally identifiable information. Once the user submits the form, the data is transmitted directly to the attacker. At this stage, the attacker collects the harvested credentials and stores them for malicious use. These details may then be used to log into the victim's real accounts on authentic websites. With access to genuine platforms, the attacker can carry out unauthorized transactions, steal additional information, or propagate further attacks.

In the final step, the attacker uses the stolen data to exploit the victim's digital identity. This may involve financial theft, identity fraud, or corporate espionage, depending on the attack's intent and the nature of the compromised information. This visual representation underscores the importance of proactive security measures and user awareness. The deceptive simplicity of the attack highlights why technological solutions—such as machine learning-based phishing detection—are vital in identifying and neutralizing such threats in real-time.


1.
Attacker sends phishing mail to target
Hacker
Target
2.
Victim clicks on Phishing link and visits fake website
3.
Hacker collects important credentials
4.
Hacker uses victim's credentials to access private information
Original Website
Phishing Website

2. Methodology



The phishing detection system proposed in this study is based on an experimental framework that utilizes supervised machine learning techniques. Among the various models available, a Decision Tree classifier is employed due to its interpretability and effectiveness in classification tasks. This model is trained to distinguish between legitimate and phishing websites by analyzing specific features extracted from the website URLs and associated metadata.

The dataset used for training is publicly available and includes various attributes that are indicative of phishing behavior. Key features such as the presence of IP addresses in the URL, use of special characters, length of the URL, and whether HTTPS is enabled are considered. Before model training, the data is carefully preprocessed through cleaning, feature extraction, and scaling to improve the overall prediction accuracy and reduce noise or inconsistencies within the dataset.

For model evaluation, the dataset is divided into training and testing subsets, with 80% of the data used to train the model and 20% reserved for validation. To further ensure that the model's performance is consistent and not overfitted to a particular data segment, k-fold cross-validation is applied. This validation technique helps assess the model's generalization capability and robustness when deployed in real-world scenarios.

To enhance the reliability of this evaluation and mitigate the risks of overfitting, **k-fold cross-validation** is employed. In this method, the dataset is divided into *k* equal parts, and the model is trained and validated *k* times—each time using a different fold for testing and the remaining folds for training. This not only provides a more accurate estimate of the model's performance but also confirms its stability and ability to generalize across diverse samples. Cross-validation thus adds a layer of robustness, ensuring the model maintains high accuracy beyond the specific dataset used in development.

To make the system accessible to users, a web-based interface has been developed using React for the frontend. This platform allows users to input website URLs and receive instant classification results indicating whether a URL is suspicious or safe. The backend logic is powered by Python, while Firebase is integrated for secure and efficient data handling. The user interface emphasizes ease of use, responsiveness, and security, making the system not only functional but also practical for day-to-day use. This modular and scalable design ensures that the phishing detection tool can be adapted and enhanced over time to meet evolving cybersecurity threats.

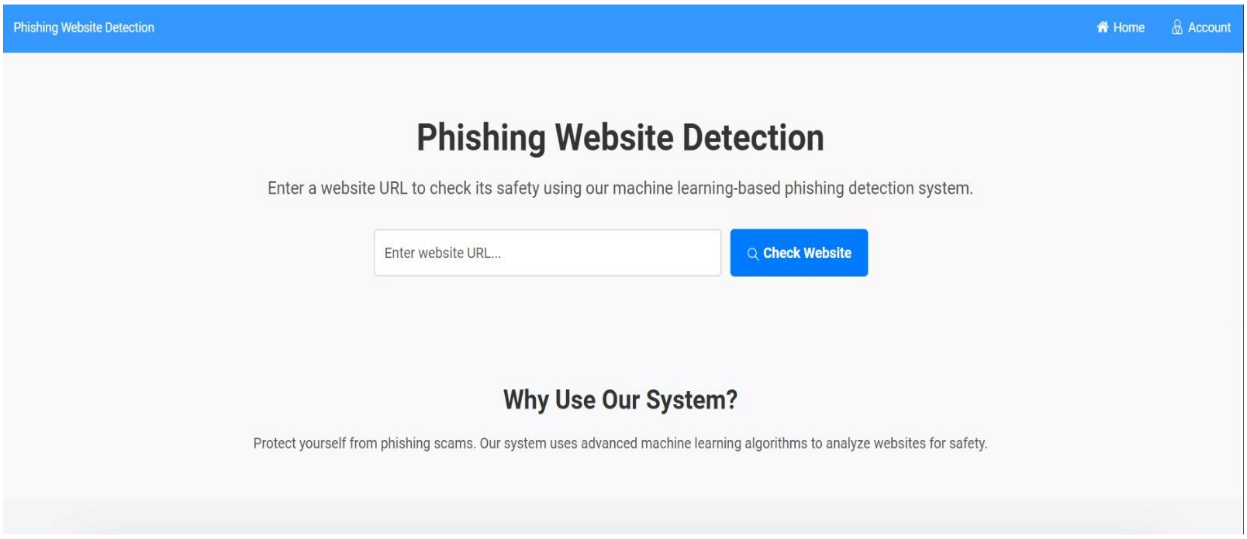
The modular and scalable architecture allows easy integration of new features and algorithms as phishing techniques evolve. This flexibility ensures long-term relevance and adaptability in dynamic cybersecurity environments.

3. Requirement specifications:

Component	Description	Recommended Requirement
Processor	Central Processing Unit (CPU) for running machine learning algorithms and the web application.	Inlet Core i7 or equivalent (hexacore)
RAM	Memory for handling data processing and model training.	8GB RAM or more
Storage	Disk space for datasets , model files, and applications storage.	512GB SSD or larger
GPU	Graphics processing unit for accelerating machine learning model training(optional)	NVIDIA RTX 2060 or higher
Network	Internet connectivity for data collection and model development	25 Mbps download speed or higher

Software	Description	Version / Specification
Operating system	The platform on which the project will run.	Windows 10/11, Ubuntu 20.04 or later, or macOS
Python	Programming language for machine learning model development.	Python 3.7 or later
Machine Learning Libraries	Libraries for implementing machine learning algorithms.	-Scikit-learn,-TensorFlow/ Keras,-NumPy,-Pandas,-Matplotlib
Web Framework	Freamework for building the web application.	Flask (for python backend)
Frontend Library	JavaScript library for building the user interface.	React.js (latest stable version)
Database management system	Database for storing user data and applications state.	Firebase

4. Screenshots



Phishing Website Detection

Enter a website URL to check its safety using our machine learning-based phishing detection system.

Why Use Our System?

Protect yourself from phishing scams. Our system uses advanced machine learning algorithms to analyze websites for safety.

Fig 1: Homepage

Fig1. Homepage Overview: Phishing Website Detection Interface

The image above displays the **homepage** of a phishing detection web application that leverages machine learning to identify and classify suspicious websites. Designed with simplicity and clarity in mind, the interface allows users to easily input a website URL and instantly verify its safety. This tool serves as a practical solution to combat phishing by offering real-time analysis based on learned patterns from a trained model.

At the center of the interface is a text input field where users can paste or type the URL they wish to examine. Accompanying this field is a **"Check Website"** button, which triggers the backend system to analyze the entered URL using a machine learning algorithm. Upon evaluation, the system determines whether the site is legitimate or potentially malicious, based on several key features extracted from the URL and metadata.

Beneath the input section, the website outlines the **purpose of the tool** under the heading “Why Use Our System?” This section communicates the core value proposition: protecting users from phishing scams through advanced, data-driven detection mechanisms. By utilizing machine learning algorithms, the system continuously improves its ability to detect emerging threats that traditional methods might overlook.

Additionally, the top navigation bar includes **"Home"** and **"Account"** options, suggesting that the platform may offer personalized features or user account management. The clean and responsive design emphasizes user-friendliness, making the tool accessible even to those with limited technical knowledge. This front-end, developed using React, is tightly integrated with a Python backend and Firebase database, offering a seamless experience that balances functionality and ease of use.

Fig2. Analyzed Website Feature Summary

The system has computed the **URL length as 52 characters**, and identified specific characteristics that may indicate whether the site is phishing or legitimate. Each feature is assigned a value of **1 (present/high risk), 0 (neutral/absent), or -1 (absent/low risk)**. This numerical representation helps the machine learning model in evaluating the overall legitimacy of the URL. Referring to fig 2.

Key Features and Their Interpretations:

UsingIP=1
Indicates that an IP address might be involved in the URL, which is a common tactic in phishing attempts.
LongURL=1
A longer-than-usual URL is often used to obscure malicious intent, suggesting a potential red flag.
ShortURL=1
The URL might be masked using a shortening service, which can hide the true destination and increase risk.
Symbol@=1
The presence of an “@” symbol is suspicious, as it can mislead users by redirecting to a different domain.
Redirecting//=1
The occurrence of multiple slashes (‘//’) in unusual positions may indicate a redirection attempt, often used in phishing.
PrefixSuffix-=1
If hyphens are used to mimic legitimate domain names (e.g., face-book.com), it can signal impersonation.

SubDomains=0

A neutral value, indicating the number of subdomains is within a normal range—not necessarily suspicious.

HTTPS=1

Despite being a positive indicator of security, phishing sites may also use HTTPS. Hence, it alone isn't sufficient for classification.

DomainRegLen=-1

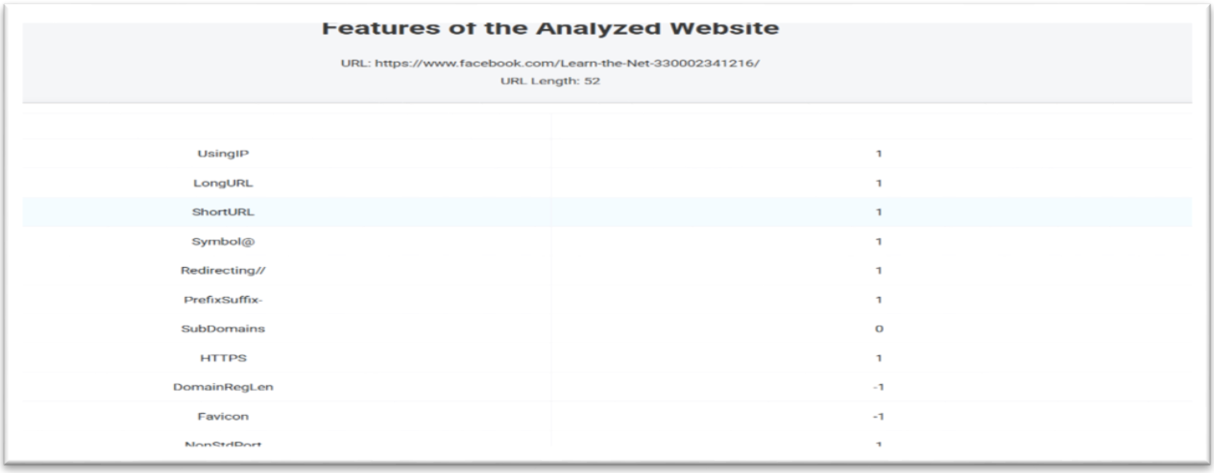
A negative value indicates that the domain has been registered for a longer duration, which generally correlates with legitimacy.

Favicon=-1

A standard favicon is present, which is typical of trusted websites and may reduce suspicion.

NonStdPort=1

The use of non-standard ports can be indicative of unusual behavior or attempts to bypass security protocols.



Features of the Analyzed Website

URL: https://www.facebook.com/Learn-the-Net-330002341216/

URL Length: 52

UsingIP	1
LongURL	1
ShortURL	1
Symbol@	1
Redirecting//	1
PrefixSuffix-	1
SubDomains	0
HTTPS	1
DomainRegLen	-1
Favicon	-1
NonStdPort	1

Purpose and Use

These extracted features are used as input to the phishing detection model, such as a Decision Tree classifier. The model evaluates patterns across multiple features to classify the URL as legitimate or phishing. This kind of analysis enhances detection accuracy by considering both surface-level attributes (like HTTPS) and hidden structural patterns (like redirects or short URLs), ultimately contributing to more reliable, data-driven cybersecurity defenses.

5. Conclusion

This project successfully implemented a reliable phishing website detection system using the Decision Tree classification algorithm, attaining a commendable accuracy rate of 85%. The architecture integrates modern technologies including React for the front-end interface, Python for backend logic, and Firebase for secure and scalable data management. This combination delivers a responsive, robust, and extensible platform for real-time threat assessment.

To validate the system's performance, extensive evaluation was conducted using multiple model assessment techniques. The Decision Tree model was benchmarked against other prominent classifiers such as Random Forest and Deep Neural Networks. Among these, the Decision Tree demonstrated the best balance of accuracy and computational efficiency for the given dataset and use case. In addition, the inclusion of an admin panel adds depth to the system, enabling monitoring of detection history and generation of visual analytics through charts and graphs, including bar plots, histograms, and heatmaps.

The system blends advanced machine learning techniques with a user-centric design to address the growing threat of phishing attacks. Its real-time analysis capability empowers users to verify website safety quickly and reliably. The intuitive interface ensures accessibility even for non-technical users, thereby increasing the tool's practical utility.

Looking ahead, future enhancements could involve enlarging the dataset to improve model generalization and exploring additional machine learning models or hybrid approaches to boost detection accuracy and resilience. These improvements will further solidify the platform's effectiveness in tackling evolving cyber threats.

REFERENCES

[1] R. Devakunchari, "Analysis on big data over the years," International Journal of Scientific and Research Publication, vol. 04, no. 01, Jan 2014.

[2] Larry Sanger, "Crime," en.wikipedia.org/wiki/Crime, Sep 20, 2001

[3] Andrews, "Cybercrime," http://en.wikipedia.org/wiki/Computer_crime , Oct 15, 2003

[4] "Cyber," merriam-webster.com/dictionary/cyber, Jan 2021.

[5] Sharma, Ushamary and Ghisingh, Seema and Ramdinmawii, Esther, “A Study on the Cyber-crime and Cyber Criminals: A Global Problem,” International Journal of web Technology, vol.03. pp.172-179, June 2014.