International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 14, Issue 2, 2026

Phishing Detection System Using Machine Learning

P.Ganesh Kumar[1], Kallepalli Bhavyasri[2], Karnakota Hamsika[3], Kattoju Lakshmi Chetana[4]
[1]Assistant Professor; Department of Information Technology Bhoj Reddy Engineering College for Women Hyderabad India
[2,3,4]B.Tech Students; Department of Information Technology Bhoj Reddy Engineering College for Women Hyderabad India
Mail Id; kallepallibhavyasri@gmail.com[2],karnakotahamsika@gmail.com[3],lakshmichetana11@gmail.com[4]

Abstract
Phishing attacks represent one of the most common and damaging forms of cybercrime, where malicious actors deceive users into revealing confidential information through fraudulent websites and deceptive URLs. As phishing techniques continue to evolve, traditional detection mechanisms struggle to effectively identify these threats. This study proposes a machine learning–based framework for detecting phishing URLs by analyzing multiple URL characteristics. The system focuses on extracting meaningful features, including lexical attributes, domain-related information, and statistical indicators that distinguish legitimate URLs from malicious ones. Several machine learning algorithms were implemented and evaluated, such as Decision Tree, Support Vector Machine (SVM), Random Forest, and ensemble-based approaches. Experimental results indicate that the Gradient Boosting classifier outperformed other models, achieving an accuracy of 97% in phishing URL classification. The findings emphasize the significance of effective feature engineering and appropriate model selection in improving detection performance. The proposed approach offers an automated and reliable solution for early identification of phishing threats, contributing to enhanced cybersecurity protection for users and online platforms.

Keywords
Phishing Detection, Machine Learning, Cybersecurity, URL Classification, Gradient Boosting, Feature Engineering, Ensemble Learning, Malicious URL Detection.

Introduction

Phishing has become one of the most prevalent cyber threats in modern digital environments. Attackers design deceptive websites and URLs that imitate legitimate platforms in order to trick users into disclosing sensitive information such as usernames, passwords, banking credentials, or personal data. These attacks are commonly delivered through fraudulent emails, social engineering messages, or fake web pages that appear authentic to unsuspecting users. As internet usage continues to expand across personal, educational, and commercial sectors, the impact of phishing incidents has also increased significantly.The growing sophistication of phishing techniques makes it increasingly difficult for users to distinguish between legitimate and malicious web addresses. Cybercriminals frequently modify URL structures, domain names, and webpage designs to closely resemble trusted websites. As a result, traditional user awareness alone is often insufficient to prevent such attacks. This situation highlights the necessity of developing automated detection mechanisms capable of identifying phishing URLs quickly and accurately before users interact with them.To address this challenge, this work proposes a web-based phishing detection system that allows users to check whether a given URL is legitimate or potentially malicious. The system employs machine learning techniques to analyze different URL characteristics and classify them accordingly. By integrating an intuitive interface with an intelligent detection model, the application aims to enhance online safety and reduce the risk of phishing-related security breaches for both individuals and organizations.

Existing System

Earlier phishing detection approaches primarily relied on blacklist-based methods. These systems maintain databases containing known phishing URLs and compare user-submitted URLs against these lists. Although blacklist techniques are straightforward and relatively easy to implement, they suffer from several limitations. Most importantly, they can only detect URLs that have already been identified and recorded in the database. Newly created phishing websites, which often appear and disappear rapidly, remain undetected until they are added to the blacklist.The rapid evolution of phishing strategies has reduced the effectiveness of purely blacklist-driven solutions. Attackers frequently generate new domain names or slightly modify existing URLs, enabling them to bypass traditional detection systems. Consequently, these methods often experience reduced detection accuracy and may produce a higher number of false positives, which can negatively affect system reliability and user trust.

To overcome these issues, modern phishing detection techniques increasingly focus on analyzing the intrinsic properties of URLs. Feature-based methods examine various attributes such as URL length, use of special symbols, presence of secure protocols (HTTPS), domain registration information, and structural patterns within the URL.

By evaluating these characteristics, the system can detect suspicious patterns even when the URL is not present in existing phishing databases.

Proposed System

The proposed system introduces an intelligent phishing URL detection platform that utilizes machine learning algorithms combined with comprehensive feature extraction techniques. The primary goal of the system is to enhance user protection by automatically identifying suspicious URLs and distinguishing them from legitimate ones through data-driven analysis.When a user submits a URL through the web interface, the system initiates a feature extraction process that evaluates multiple characteristics of the input URL. These features include parameters such as URL length, usage of HTTPS protocols, presence of unusual characters (for example @ symbols, double slashes, or IP-based URLs), number of subdomains, and occurrence of keywords frequently associated with phishing attacks. The extracted attributes form a feature vector that is forwarded to the machine learning classification module.

Literature Review

Survey

A systematic review of existing studies was conducted to understand the current progress in phishing URL detection and to identify potential research gaps. The objective of this survey was to examine previously proposed techniques, evaluate their strengths and limitations, and determine promising research directions for improving phishing detection systems. By analyzing multiple studies in the domain of cybersecurity and machine learning, the survey highlights the methodologies that have been widely adopted for detecting malicious URLs.Developing a structured literature review requires the formulation of clear research questions that guide the investigation process. These questions serve as the foundation for analyzing existing studies and identifying areas where further improvements are required.

Literature on Machine Learning–Based Phishing Detection

In recent years, machine learning techniques have been extensively explored as effective solutions for detecting phishing attacks in online environments. Researchers have focused on developing intelligent systems capable of identifying fraudulent websites, malicious emails, and deceptive URLs by analyzing patterns commonly associated with phishing activities.Most machine learning–based approaches rely on classification algorithms to distinguish between legitimate and malicious entities. Algorithms such as Decision Trees, Random Forest, Support Vector Machines (SVM), and Artificial Neural Networks are frequently used to build predictive models for phishing detection. These models are trained using datasets that contain labeled examples of phishing and legitimate URLs.

Literature on Phishing Detection Systems

Various techniques have been proposed in the literature for identifying phishing attacks, ranging from traditional detection mechanisms to more advanced intelligent approaches. Early detection systems mainly relied on blacklist-based techniques and rule-based filtering. In these methods, previously identified phishing URLs or suspicious patterns are stored in databases and compared against incoming URLs to determine their legitimacy.

Although these methods are simple to implement, they are limited in their ability to detect newly generated phishing websites. Since blacklist systems depend on previously reported attacks, they cannot effectively identify zero-day phishing threats that have not yet been recorded. This limitation significantly reduces their effectiveness in rapidly evolving threat environments.Several machine learning algorithms have been successfully applied in phishing detection systems, including Decision Trees, Random Forest, Support Vector Machines (SVM), and Naive Bayes classifiers. These models have demonstrated improved detection performance compared to traditional approaches, particularly when combined with effective feature extraction and dataset preprocessing techniques.

Feature Extraction

The proposed phishing detection framework operates through two major phases: the training phase and the testing phase. Feature extraction plays a critical role in both phases, as it converts raw URLs into structured numerical representations that can be processed by machine learning algorithms.

Training Phase

During the training phase, raw URLs collected from datasets are analyzed to extract relevant features that help distinguish phishing websites from legitimate ones. This process involves examining different properties of the URL, such as its length, presence of special symbols, use of HTTPS protocols, number of subdomains, and other structural characteristics.

Testing Phase

After the training process is completed, the trained models are evaluated using a separate testing dataset containing previously unseen URLs. The same feature extraction process is applied to these URLs to generate feature vectors, which are then provided as input to the trained classifiers.The models predict whether each URL is legitimate or phishing based on the learned patterns. To assess the effectiveness of the detection system, several performance metrics are calculated, including accuracy, precision, recall, and F1-score. These evaluation metrics provide insights into how well the model detects phishing URLs while minimizing false positives and false negatives.

System Architecture

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 14, Issue 2, 2026

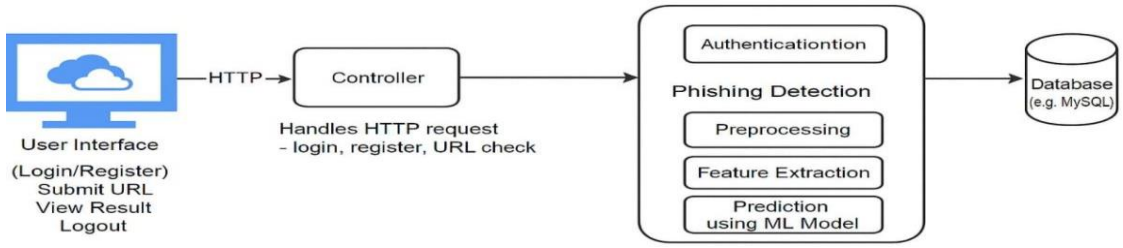


Fig. 1 System Architecture

System architecture describes the overall structure, organization, and behavior of a technological system. It represents how different components of an application interact with each other to achieve the intended objectives. In general, system architecture provides a high-level overview of the system workflow and demonstrates how information flows between different modules.The primary goal of system architecture design is to develop a comprehensive and well-structured solution that aligns with the functional and non-functional requirements of the system. It establishes the fundamental principles, concepts, and relationships among system components so that they operate coherently to fulfill the desired mission and stakeholder expectations.A well-defined architecture focuses on the conceptual and logical structure of the system rather than its detailed implementation. It emphasizes the relationships between system elements, identifies key modules, and outlines the interactions necessary for system functionality. In the context of the proposed phishing detection system, the architecture illustrates how user input, feature extraction, machine learning models, and prediction outputs are integrated to provide an effective URL classification mechanism.

Technical Architecture

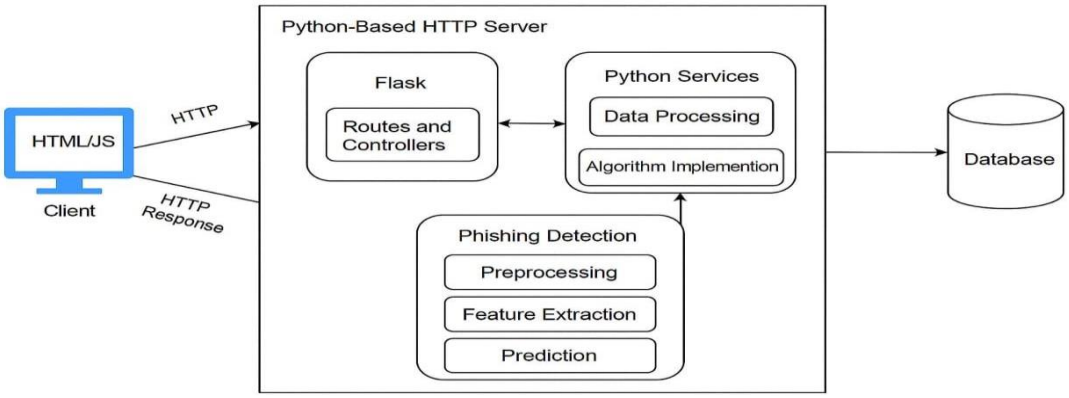


Fig.2 Technical Architecture

Technical architecture refers to the design framework that defines how hardware and software components interact to support the system's functionality. It focuses on translating application requirements into a technological infrastructure that can efficiently support system operations.The technical architecture integrates components from the application layer with the necessary technological resources such as servers, databases, programming frameworks, and machine learning libraries. These components are typically obtained from existing platforms and configured together to build a reliable computing environment.

Data Flow Diagram (DFD)

A Data Flow Diagram (DFD) is a graphical representation used to illustrate how data moves within a system and how it is processed. It visually describes the flow of information from input sources through various processing stages until the final output is produced. Because of its simple and intuitive structure, the DFD is sometimes referred to as a bubble diagram.DFDs are widely used in system modeling to represent key components of a system, including processes, data storage, external entities, and data flows. These diagrams help illustrate how information enters the system, how it is transformed through different operations, and how results are delivered to users or external systems.The primary objective of a DFD is to clearly depict how data is processed and transformed throughout the system. It shows the sequence of operations applied to the data as it travels from the input stage to the final output stage. DFDs can also be developed at multiple levels of abstraction. A high-level diagram provides a general overview of the system, while lower-level diagrams offer more detailed insights into individual processes and interactions.

Algorithms Used in the System

Gradient Boosting

Gradient Boosting is an ensemble learning method widely used for classification and regression tasks. It constructs a strong predictive model by combining

multiple weak learners, typically decision trees. The models are trained sequentially, where each new model attempts to reduce the errors made by the previous models. By iteratively improving predictions, Gradient Boosting achieves high accuracy in complex datasets.

XGBoost

Extreme Gradient Boosting (XGBoost) is an optimized implementation of the gradient boosting algorithm designed for high performance and computational efficiency. It improves model speed and scalability by using advanced optimization techniques such as parallel processing and regularization. XGBoost builds a series of decision trees where each tree attempts to minimize a predefined loss function.

Logistic Regression

Logistic Regression is a statistical classification technique used to predict binary outcomes. It estimates the probability that a given input belongs to a specific class using the logistic (sigmoid) function. In phishing detection systems, logistic regression can be used to estimate the likelihood that a URL is either legitimate or malicious based on extracted features.

Random Forest

Random Forest is an ensemble learning algorithm that generates multiple decision trees during the training process. Each tree independently predicts the output, and the final result is determined by combining the predictions of all trees, typically through majority voting. This method improves accuracy and reduces the risk of overfitting.

Support Vector Machine (SVM)

Support Vector Machine is a supervised machine learning algorithm commonly used for classification and regression tasks. It works by identifying the optimal hyperplane that separates data points belonging to different classes with the maximum margin. SVM is particularly effective for high-dimensional datasets and has been widely applied in phishing detection research.

System Modules

A module represents an independent functional component of a software system. Each module performs a specific task and can operate either independently or in collaboration with other modules. The proposed phishing detection system consists of the following modules:

Phishing Detection Module

The phishing detection module is responsible for classifying URLs using trained machine learning models. It evaluates the extracted features and predicts whether the given URL is legitimate or phishing. By learning patterns from previously labeled datasets, the model can identify suspicious links and warn users about potential online threats.

Implementation

Libraries Used

The implementation of the phishing URL detection system relies on several Python libraries that support machine learning, data processing, web communication, and domain analysis. These libraries play an important role in building and deploying the detection framework.

Scikit-learn

Scikit-learn is a widely used open-source machine learning library developed for the Python programming language. It provides a collection of efficient tools for data analysis, predictive modeling, and statistical learning. The library is built on top of foundational scientific computing packages such as NumPy, SciPy, and Matplotlib. Scikit-learn supports various machine learning algorithms, including classification, regression, clustering, and dimensionality reduction techniques. Due to its ease of use and extensive documentation, it is commonly adopted in both academic research and industrial applications for building and evaluating predictive models.

Flask

Flask is a lightweight web framework written in Python that facilitates the development of web applications and APIs. It follows a minimalistic design philosophy that allows developers to create scalable and flexible web services with minimal configuration. Flask provides essential features such as URL routing, template rendering, and request handling, enabling rapid development of web-based interfaces. In the proposed system, Flask is used to create the web interface through which users can submit URLs and obtain phishing detection results.

NumPy

NumPy is a core Python library designed for numerical and scientific computing. It offers powerful multidimensional array objects along with a variety of mathematical functions for performing complex computations efficiently. NumPy supports operations such as linear algebra, statistical calculations, Fourier transforms, and random number generation. Because of its optimized performance and memory efficiency, NumPy serves as the foundation for many scientific computing libraries used in machine learning applications.

Pandas

Pandas is an open-source data analysis library that provides powerful data structures for handling structured data. It enables efficient data manipulation, cleaning, and transformation using structures such as DataFrames and Series. Pandas simplifies tasks related to data loading, preprocessing, filtering, and aggregation. In machine learning workflows, it is widely used for preparing datasets before training predictive models.

Google Search Python Package

The Google Search Python package enables programmatic interaction with search engine results. It allows the system to check whether a given URL appears in search engine indexes. This functionality

can help determine whether a domain is widely recognized or suspicious, thereby contributing additional information to the phishing detection model.

Pseudocode and Feature Extraction Process

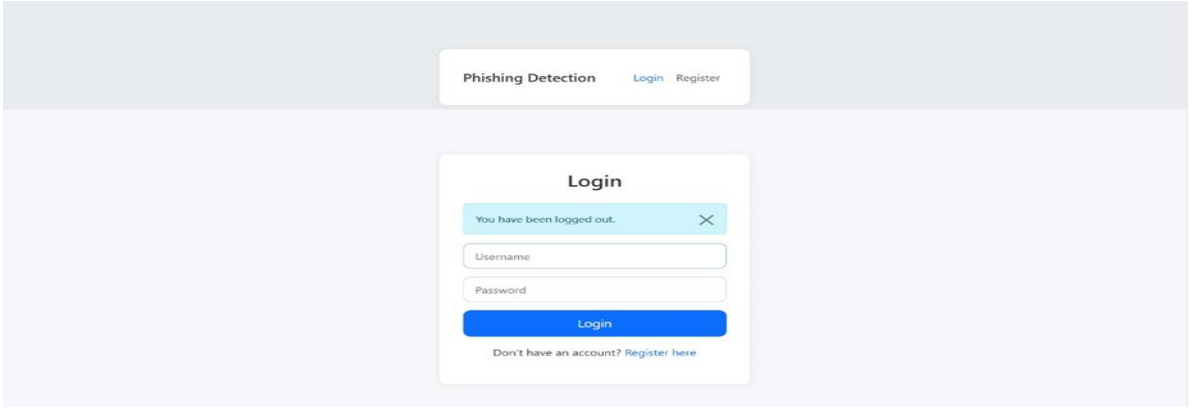
Feature extraction is a critical component of the phishing detection system. It converts raw URL inputs into structured numerical representations that can be used by machine learning algorithms. The system analyzes multiple characteristics of a URL to determine whether it exhibits suspicious behavior commonly associated with phishing attacks.The feature extraction process begins by receiving a URL as input. The system then performs several preprocessing operations, including retrieving webpage content, parsing domain information, and querying domain registration data. Various features are extracted from the URL and its associated webpage, which are later stored in a feature vector for classification.

The general procedure followed by the feature extraction module can be summarized as follows:

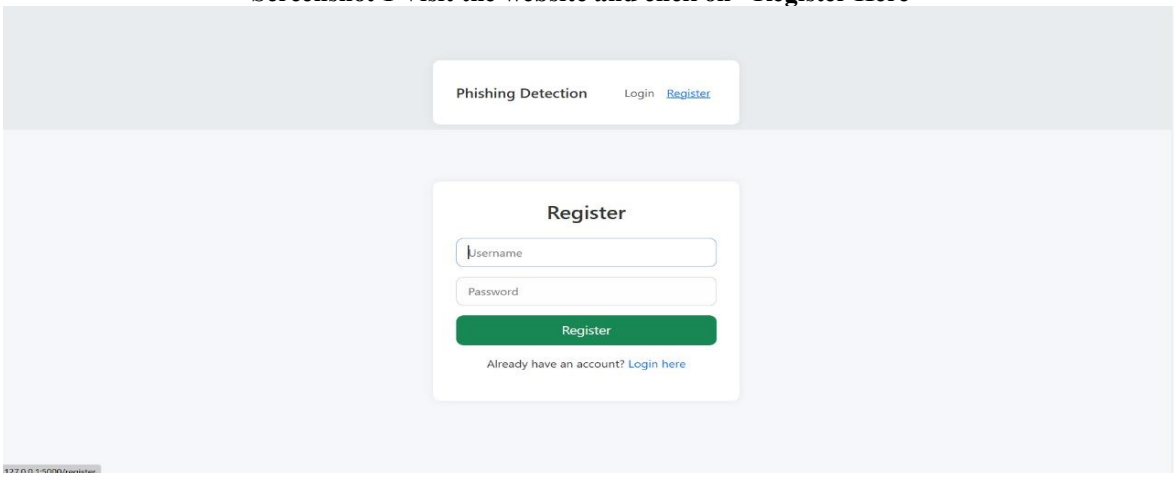
1. **InputURLProcessing**
   The system receives a URL from the user interface and initializes variables required for analysis. The URL is parsed to extract domain information and other structural components.
2. **WebpageRetrieval**
   Using HTTP requests, the system attempts to access the webpage associated with the URL. If the request is successful, the HTML content is parsed using Beautiful Soup to analyze webpage elements.
3. **DomainInformationExtraction**
   WHOIS queries are performed to obtain domain registration details such as creation date, expiration date, and ownership information. These attributes help evaluate the credibility of the domain.
4. **FeatureVectorConstruction**
   Each extracted attribute is converted into a numerical value representing whether the feature indicates legitimate or suspicious behavior. These values are stored in a feature vector.
5. **ModelInputPreparation**
   The generated feature vector is provided as input to machine learning models trained during the training phase. The model then predicts whether the given URL is phishing or legitimate.

   This feature extraction framework enables the system to analyze multiple characteristics of URLs and webpages, allowing machine learning models to detect phishing attempts with improved accuracy.

Screenshots

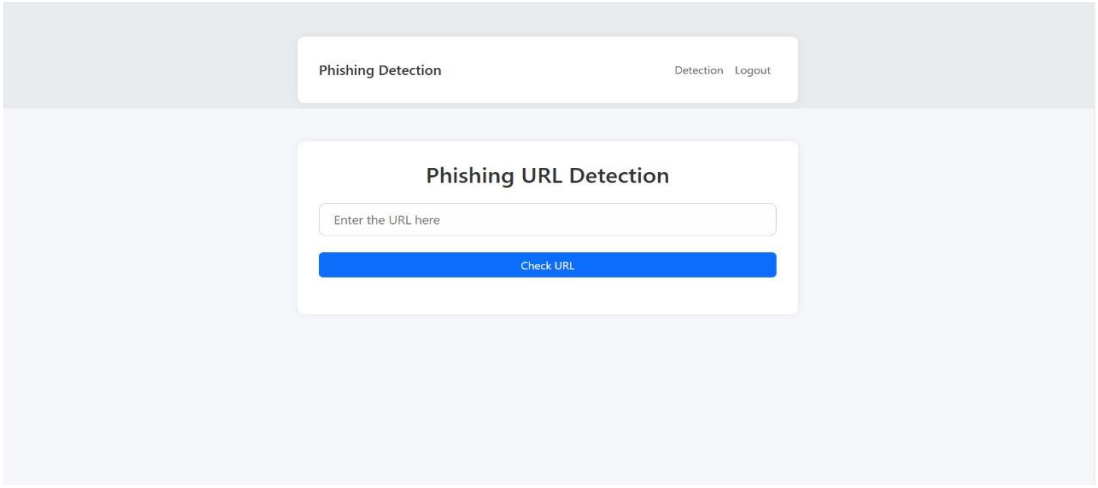


Screenshot 1 Visit the website and click on "Register Here"

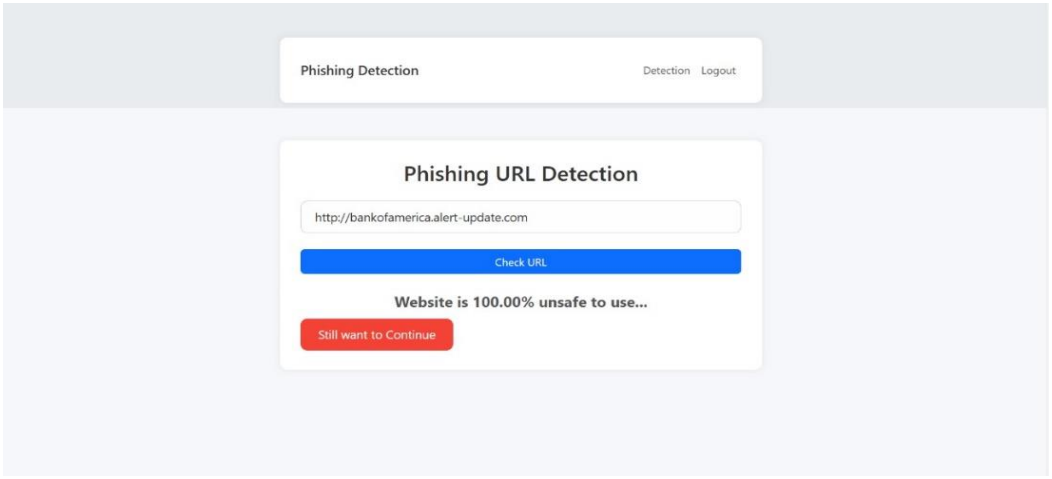


Screenshot 2 Register yourself on website giving Username and Password

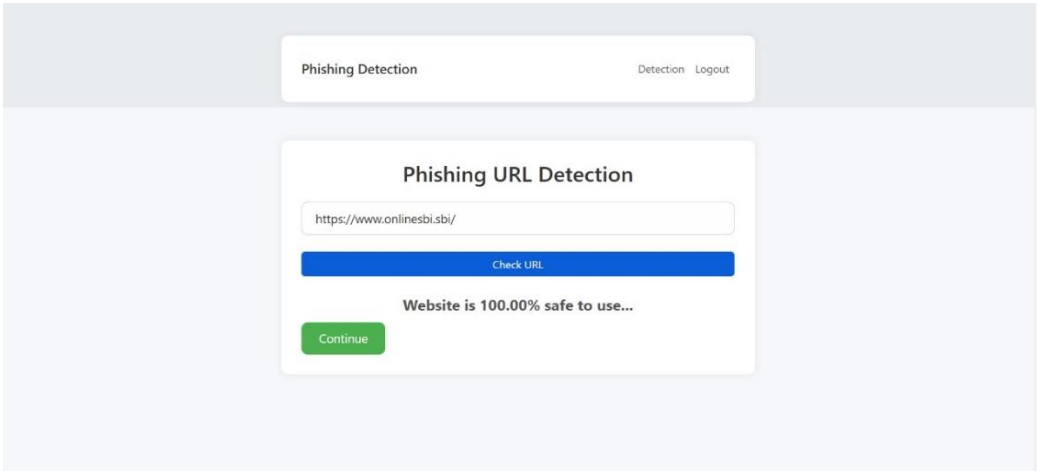
International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 14, Issue 2, 2026



Screenshot 6.3 Enter URL



Screenshot 4 Result shows URL is not safe to use



Screenshot 5 Result shows URL is safe to Use.

Conclusion

This research presents the development of an intelligent phishing detection system that leverages machine learning techniques to identify malicious websites. The proposed approach analyzes several structural characteristics of URLs to determine whether a given link is legitimate or potentially fraudulent.Multiple machine learning algorithms were evaluated during the experimentation phase to determine the most effective model for phishing detection. Among the tested models, the **XGBoost classifier achieved the highest performance with an accuracy of approximately 95%**, demonstrating its effectiveness in detecting phishing URLs. The results indicate that machine learning-based solutions can significantly enhance the detection of online threats compared to traditional rule-based methods.The developed system provides an automated and efficient mechanism for identifying phishing websites at an early stage. By

analyzing URL patterns and related attributes, the system helps reduce the risk of users interacting with malicious links. Consequently, the proposed framework contributes to improving cybersecurity awareness and protecting individuals and organizations from phishing attacks.

Future Scope

Although the current system effectively detects phishing URLs, several improvements can be considered for future development to enhance its capabilities and usability.One potential extension is the integration of educational features that inform users about phishing techniques and provide guidance on identifying suspicious online activities. Such features can increase user awareness and strengthen overall cybersecurity practices.Another promising direction involves expanding the detection framework to analyze phishing attempts in emails and messaging platforms. By incorporating natural language processing and message analysis techniques, the system could identify phishing links distributed through communication channels.The development of mobile applications for platforms such as Android and iOS would also improve accessibility, allowing users to verify suspicious links directly from their smartphones. In addition, implementing real-time alert mechanisms could provide immediate warnings and personalized security recommendations based on user activity.

References

[1] S. Parekh, D. Parikh, S. Kotak, and S. Sankhe, "A New Method for Detection of Phishing Websites Using URL Detection," IEEE Conference Proceedings, 2018, ISBN: 978-1-5386-1974-2.
[2] IEEE, "Feature Selection Techniques for Machine Learning-Based Phishing Website Detection," IEEE Conference Proceedings, 2017.
[3] IEEE Xplore Digital Library, "Phishing Website Detection Research Article," Available: http://ieeexplore.ieee.org/document/8090317
[4] S. Mohanty, "Predicting Phishing URLs Using Filter-Based Univariate Feature Selection Techniques," IEEE Conference Proceedings, 2022.
[5] V. S. Lakshmi and M. S. Vijaya, "Efficient Prediction of Phishing Websites Using Supervised Learning Algorithms," *Procedia Engineering*, vol. 30, pp. 798–805, 2012.
[6] H. A. Shaik, "Phishing URL Detection Using Machine Learning Methods," ResearchGate Publication, January 2022.
[7] D. Sahoo, "Malicious URL Detection Using Machine Learning: A Survey," 2022.
[8] U. Shetty, A. Patil, and M. Mohana, "Malicious URL Detection and Classification Using Machine Learning Models," IEEE Conference Proceedings, 2023, ISBN: 978-1-6654-7451-1.
[9] A. Garje, N. Tanwani, S. Kandale, T. Zope, and S. Gore, "Detecting Phishing Websites Using Machine Learning," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 9, no. 11, pp. 2320–2882, 2021.
[10] R. Verma, K. Shashidhar, and N. Hossain, "What's in a URL: Fast Feature Extraction and Malicious URL Detection," *Proceedings of the Seventh ACM Conference on Data and Application Security and Privacy*, pp. 55–63, 2017.