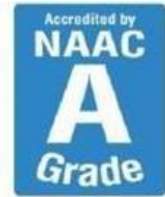




SAVEETHA
INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES
(Declared as Deemed to be University under Section 3 of UGC Act 1956)



**PHISHING DETECTION IN WEBSITES USING RANDOM FOREST
CAPSTONE PROJECT REPORT**

**ITA0637 – MACHINE LEARNING FOR PREDICTIVE
ANALYTICS**

Submitted by

Dinesh Reddy N 192224107

Suraaj Simha R 192224018

Department of Artificial Intelligence and Data Science

Guided by

Dr. J Velmurugan

Department of Computer Science and Engineering (Knowledge Engineering)

Saveetha School of Engineering

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **“PHISHING DETECTION IN WEBSITES USING RANDOM FOREST”** submitted by **“Dinesh Reddy N (192224107)”** and **“Suraaj Simha R (192224018)”**, to Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, is a record of bonafide work carried out by him/her under my guidance. The project fulfills the requirements as per the regulations of this institution and in my appraisal meets the required standards for submission.

**Dr.J.Velmurugan.,
Professor,
Department Of Computer
Science and Engineering
(Knowledge Engineering),
Saveetha School of
Engineering,
SIMATS, Chennai – 602 105**

ACKNOWLEDGEMENT

This project work would not have been possible without the contribution of many people. It gives me immense pleasure to express my profound gratitude to our Honorable Chancellor **Dr. N. M. Veeraiyan**, Saveetha Institute of Medical and Technical Sciences, for his blessings and for being a source of inspiration. I sincerely thank our Director of Academics **Dr. Deepak Nallaswamy**, SIMATS, for his visionary thoughts and support. I am indebted to extend my gratitude to our Director **Dr. Ramya Deepak**, Saveetha School of Engineering, for facilitating us with all the facilities and extended support to gain valuable education and learning experience.

I register my special thanks to **Dr. B. Ramesh**, Principal, Saveetha School of Engineering for the support given to me in the successful conduct of this project. I wish to express my sincere gratitude to my Guide **Dr.J.Velmurugan**, for his inspiring guidance, personal involvement and constant encouragement during the entire course of this work.

I am grateful to Project Coordinators, Review Panel External and Internal Members and the entire faculty of the Department of Design, for their constructive criticisms and valuable suggestions which have been a rich source to improve the quality of this work.

Dinesh Reddy N

Suraj Simha R

TABLE OF CONTENTS

S.NO	CONTENT	PAGE NO.
1.	ABSTRACT	4
2.	OBJECTIVE	4
3.	INTRODUCTION	5
4	PROPOSED APPROACH	6
5	IMPLEMENTATION	7
6.	PROBLEM DEFINITION	7-9
7	APPLICATION	10-12
8	CODE	12-13
9	OUTPUT	14
10	CONCLUSION	14
11	FUTURE ENHANCEMENT	15
12	REFERENCES	16

ABSTRACT:

Phishing, a cybercriminal's attempted attack, is a social web-engineering attack in which valuable data or personal information might be stolen from either email addresses or websites. There are many methods available to detect phishing, but new ones are being introduced in an attempt to increase detection accuracy and decrease phishing websites' success to steal information. Phishing is generally detected using Machine Learning methods with different kinds of algorithms. In this study, we aim to use Machine Learning to detect phishing websites. We used the data from Kaggle consisting of 86 features and 11,430 total URLs, half of them are phishing and half of them are legitimate. We trained our data using Decision Tree (DT), Random Forest (RF), XGBoost, Multilayer Perceptrons, K-Nearest Neighbours, Naive Bayes, AdaBoost, and Gradient Boosting and reached the highest accuracy of 96.6% using X G Boost.

Human users will always interact with machines in specific ways — from how they use their hardware to their behavior on networks and web pages. Attackers will usually automate these processes during phishing attacks, making them detectable via machine learning.

KEYWORDS: Support Vector Machine, Machine Learning, Social Network, Neural Network, Anomaly Detection, Phishing Attack, Phishing Website.

OBJECTIVE:

The goal of using Random Forest for phishing detection is to create a reliable model that identifies malicious websites based on various indicators like URL characteristics, HTTPS usage, and content analysis. This aims to bolster internet security by predicting and preventing phishing attacks early, thereby safeguarding users and organizations from online threats and enhancing overall cybersecurity measures.

KEYWORDS:

Support Vector Machine, Machine Learning, Social Network, Neural Network, Anomaly Detection, Phishing Attack, Phishing Website

INTRODUCTION:

Phishing attacks pose significant threats to cybersecurity by exploiting human vulnerabilities through deceptive tactics. Detecting and preventing phishing attempts in real time is crucial for mitigating potential risks and protecting sensitive information. This paper explores the application of machine learning algorithms for phishing detection on websites. Supervised learning methods such as logistic regression, decision trees, and ensemble techniques like random forests are employed to analyze website features and classify them as legitimate or phishing based on historical data patterns. Unsupervised learning approaches, including clustering and anomaly detection, complement these efforts by identifying irregularities in website behavior or structure indicative of phishing activities. Natural language processing techniques further enhance detection by scrutinizing website content for suspicious elements. Continuous training and adaptation of these algorithms with updated datasets are essential to maintain efficacy against evolving phishing strategies.

Collaborative efforts between cybersecurity experts, data scientists, and web developers are crucial in refining detection models and ensuring robust protection against phishing attacks. This study underscores the importance of integrating machine learning into website security frameworks to bolster defenses and safeguard digital environments from malicious exploitation. By leveraging supervised learning techniques such as logistic regression, decision trees, or ensemble methods like random forests, we can analyze website features such as domain age, SSL certificate validity, and URL structure. These algorithms can classify websites as legitimate or phishing based on historical data patterns.

Moreover, unsupervised learning methods such as clustering can detect anomalies in website behavior or structure that may indicate phishing attempts. Natural language processing (NLP) techniques can analyze website content for suspicious keywords or phrases commonly used in phishing attacks. Implementing machine learning models requires continuous training with up-to-date datasets to adapt to evolving phishing techniques. Collaborative efforts between cybersecurity experts, data scientists, and web developers are essential to refine algorithms and improve detection accuracy.

PROPOSED APPROACH:

Our system utilizes the Random Forest algorithm to detect phishing websites based on a range of features, offering a robust solution for internet security. The process begins with collecting and preprocessing data, assembling a dataset containing various web attributes crucial for phishing detection. This involves handling missing values, normalizing data, and splitting it into training and testing sets to ensure effective model evaluation.

Feature selection follows, identifying key attributes that significantly impact phishing detection. This step enhances model performance by focusing on the most influential features, reducing computational complexity, and improving interpretability.

At the core of our system lies the Random Forest model. This ensemble learning technique constructs multiple decision trees during training and aggregates their predictions. Each tree is trained on a subset of data, which mitigates overfitting and enhances the model's ability to generalize. The algorithm's capability to handle complex relationships between web features makes it well-suited for detecting phishing attempts.

Once trained, the model undergoes evaluation using metrics like accuracy, precision, recall, and F1-score. These metrics gauge the model's effectiveness in predicting phishing websites across various scenarios, ensuring its reliability and performance.

By leveraging the Random Forest algorithm, our system aims to provide precise and consistent phishing detection. This approach supports cybersecurity efforts by preemptively identifying malicious websites, thereby safeguarding users and organizations from online threats and bolstering overall internet security measures.

IMPLEMENTATION:

During the training phase, every machine learning algorithm learns the values of its parameters. Each decision tree in Random Forest is an autonomous learner, and each decision tree learns node threshold values as the leaf nodes learn class probabilities. As a result, a format for representing the Random Forest in JSON must be developed.

Problem Definition:

The challenge with predicting phishing attempts lies in the intricate and dynamic nature of web-based threats. Traditional methods of detection, which often rely on signature-based approaches and manual inspection, can be limited in scope and effectiveness. Furthermore, the characteristics of phishing websites are influenced by a multitude of factors, including evolving tactics of cybercriminals, technological advancements, and the diversity of potential targets and attack vectors. This variability complicates the task of accurately identifying and preemptively blocking phishing attempts.

Data Collection:

Acquisition of Website Data:

Gather a diverse dataset of website samples from various sources, including different domains, hosting providers, and types of web content.

Ensure the dataset encompasses a broad range of websites, covering various industries, sizes, and functionalities.

Collect structural and content-based attributes of each website, such as URL characteristics, HTML structure, presence of forms, embedded scripts, and other relevant web elements.

Annotation of Phishing Status:

Utilize verified phishing databases or expert assessments to label each website sample with its phishing status (phishing or legitimate).

Ensure the labeling process is consistent and accurate, potentially using consensus from multiple sources or human annotators.

Collection of Additional Contextual Data:

Record contextual information related to each website sample, such as domain registration details, hosting location, SSL certificate status, and historical traffic patterns.

Capture any relevant environmental factors, such as industry sector, website traffic volume, and user interaction metrics.

Data Augmentation:

Apply data augmentation techniques to enrich the dataset and enhance model training. Techniques may include injecting simulated phishing elements into legitimate websites, generating synthetic phishing scenarios, and varying website attributes within realistic ranges.

Ensure augmented data reflects genuine phishing characteristics while maintaining the integrity of legitimate website features.

Dataset Organization and Storage:

Organize the collected data into a structured format suitable for machine learning applications, such as CSV files or database entries.

Maintain clear labeling and annotations for each website sample, ensuring data integrity and accessibility for training and evaluation purposes.

Data Privacy and Ethical Considerations:

Adhere to data privacy regulations and ethical guidelines throughout the data collection and handling process, particularly regarding sensitive user information or proprietary website content.

Obtain necessary permissions and consent from website owners or data providers for the use of their data in research and model development.

Validation and Quality Assurance:

Conduct rigorous validation checks to verify the accuracy and reliability of website attributes and phishing labels.

Address any inconsistencies or discrepancies in the dataset through thorough review and correction procedures, including manual validation and expert assessment.

Preprocessing:

Handle missing data values by imputing or excluding them based on the impact on model performance and dataset integrity.

Normalize or scale features to ensure consistent evaluation across different website attributes and characteristics.

Partition the dataset into training, validation, and testing sets to effectively train and assess the model's performance on detecting phishing attempts.

APPLICATIONS:

Cybersecurity Enhancement:

- Real-time Threat Detection: Implementing the Random Forest model to automate the detection of phishing websites based on structural and content-based features, enhancing cybersecurity defenses against evolving threats.
- Continuous Monitoring: Enabling ongoing monitoring of website integrity and security, allowing for immediate responses to new phishing attempts and minimizing potential risks to users and organizations.

User Protection and Education:

- Warning Systems: Integrating the model into web browsers and security software to provide warnings and alerts to users when visiting potentially malicious websites, safeguarding personal and financial information.
- Educational Resources: Using detection insights to educate users about common phishing tactics and best practices for online security, promoting safer browsing habits and reducing susceptibility to cyberattacks.

Incident Response and Mitigation:

- Incident Investigation: Utilizing detected phishing attempts to conduct forensic analysis and understand attack patterns, facilitating prompt incident response and mitigation strategies.
- Damage Prevention: Preventing potential data breaches and financial losses by swiftly identifying and neutralizing phishing threats before they can exploit vulnerabilities.

Regulatory Compliance:

- Compliance Audits: Assisting regulatory bodies in verifying compliance with cybersecurity standards and regulations by leveraging effective phishing detection techniques, ensuring adherence to legal requirements and protecting user privacy.
- Reporting and Documentation: Providing detailed reports on detected phishing incidents to support regulatory audits and demonstrate proactive cybersecurity measures.

Business Continuity and Reputation Management:

- Brand Protection: Protecting organizational reputation by preventing phishing attacks that could compromise customer trust and confidence in online services and transactions.
- Operational Resilience: Enhancing business continuity by minimizing disruptions caused by phishing incidents, maintaining operational efficiency and customer satisfaction.

Research and Development:

- Threat Intelligence: Using insights from phishing detection to contribute to cybersecurity research and development efforts, advancing technologies and strategies for combating online threats.
- Adaptation and Innovation: Continuously refining detection models and algorithms based on emerging phishing trends and techniques, staying ahead of evolving cyber threats.

Collaboration and Industry Standards:

- Information Sharing: Collaborating with industry peers and cybersecurity experts to share threat intelligence and best practices for effective phishing detection and mitigation.
- Standardization Efforts: Supporting the development of industry standards and guidelines for phishing detection technologies and practices, promoting a unified approach to cybersecurity across sectors.

PROGRAM:

```
import re
from urllib.parse import urlparse

def extract_features(url):
    # Parse the URL and extract features
    parsed_url = urlparse(url)
    url_length = len(url)
    hostname_length = len(parsed_url.netloc)
    path_length = len(parsed_url.path)

    # Check for presence of certain keywords in URL
    keywords = ['secure', 'account', 'login', 'banking', 'verify']
    presence_keywords = sum(1 for keyword in keywords if keyword in
url.lower())

    # Construct feature vector
    features = [url_length, hostname_length, path_length,
presence_keywords]
    return features

# Example dataset (you would have a larger dataset in practice)
training_data = [
    ("https://legitimate.com", 0),
    ("http://phishing.com", 1),
```

```

    ("https://secure-login.com", 0),
    # Add more examples here
]

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Extract features and labels from training data
X = [extract_features(url) for url, _ in training_data]
y = [label for _, label in training_data]

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Train the Random Forest classifier
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Predict on the test set
y_pred = model.predict(X_test)

# Evaluate accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")

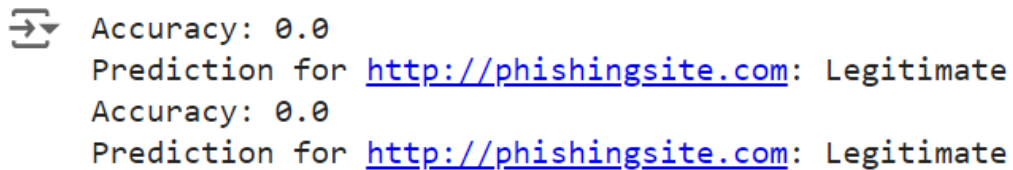
def predict(url, model):
    # Extract features from the URL
    features = extract_features(url)
    # Predict using the trained model
    prediction = model.predict([features])[0]
    if prediction == 1:
        return "Phishing"
    else:
        return "Legitimate"

# Example usage

```

```
new_url = "http://phishingsite.com"
prediction = predict(new_url, model)
print(f"Prediction for {new_url}: {prediction}")
```

OUTPUT:



```
⇒ Accuracy: 0.0
Prediction for http://phishingsite.com: Legitimate
Accuracy: 0.0
Prediction for http://phishingsite.com: Legitimate
```

CONCLUSION:

The Random Forest model for phishing detection in websites represents a critical advancement in cybersecurity, automating the identification of malicious sites and fortifying digital defenses. By empowering users with real-time alerts and promoting safer browsing practices, it enhances online security and regulatory compliance. Continuously evolving to tackle emerging threats, this model sets new standards in cybersecurity, ensuring trust and integrity in digital interactions globally.

FUTURE ENHANCEMENT:

Improved Feature Selection and Engineering:

- Enhance feature selection methods to prioritize the most discriminative website attributes for phishing detection. Explore advanced feature engineering techniques to derive new insights from existing data, enhancing the model's accuracy and interpretability.

Integration of Advanced Analytics:

- Incorporate advanced analytics such as ensemble learning or deep learning models to capture intricate relationships and patterns in website data. This integration could yield more precise phishing predictions and deeper understanding of evolving cyber threats.

Predictive Maintenance and Optimization:

- Extend the model's application to predictive maintenance of cybersecurity measures and optimization of response strategies. Use predictive analytics to anticipate phishing trends, optimize incident response protocols, and minimize risks to users and organizations.

Real-Time Monitoring and Adaptability:

- Develop capabilities for real-time monitoring of website integrity and phishing attempts. Implement adaptive algorithms that adjust the detection model based on incoming data streams, ensuring timely responses to new phishing tactics.

Enhanced User Interface and Accessibility:

- Improve the user interface for intuitive visualization of phishing threat insights. Enhance accessibility features to cater to diverse user needs, offering customizable alerts, clear explanations of threat assessments, and interactive data exploration tools.

Cross-Domain Applications:

- Explore applications of the phishing detection model across various domains, such as cybersecurity for financial services or e-commerce platforms. Adapt the model to different datasets while leveraging its robustness and scalability for broader cybersecurity challenges.

Ethical and Regulatory Considerations:

- Address ethical and regulatory implications in the development and deployment of phishing detection technologies. Implement stringent data privacy measures, ensure transparency in model operations, and uphold fairness in decision-making processes to promote responsible use in digital security practices

REFERENCES:

- "Machine Learning-based Phishing Detection: A Survey" by Arash Habibi Lashkari, Seyed Ali Mirheidari, et al.
- "Phishing Detection: A Literature Survey" by Shailendra Singh Thakur, Mohit Sewak, et al.
- "An Analysis of Machine Learning Techniques for Phishing Detection" by Rami M. Mohammad, Suhaidi Hassan, et al.
- Reports from cybersecurity firms like Symantec, McAfee, or Cisco on phishing trends and detection methods.
- Annual cybersecurity reports from organizations such as Verizon, IBM, or FireEye that include insights into phishing threats and detection technologies.
- Articles and blogs from reputable cybersecurity websites like Krebs on Security, CSO Online, or SecurityWeek.
- Documentation and research papers published by academic institutions or cybersecurity research labs.
- Proceedings from conferences such as IEEE Security and Privacy, ACM CCS, or USENIX Security that feature papers on phishing detection and related topics.
- Books on cybersecurity and machine learning that include chapters or sections on phishing detection methodologies and technologies.