International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 03 | March - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Phishing Website Detection Using Machine Learning

1st Atharva Dharrao
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
atharvadharrao11@gmail.com

2nd Divya Khandare
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
divyakhandare2003@gmail.com

3rd Akash Jadhav
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
jadhavneel19@gmail.com

4th Gaurav Mahadik
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
gauravmahadik1010@gmail.com

5th Dipika Raigar
Department of Computer Engineering
Assistant Professor, Pune Institute of Computer Technology
Pune, India
ddraigar@pict.edu

***Abstract*—Phishing websites pose a significant cybersecurity threat, deceiving users into disclosing sensitive information such as login credentials, financial details, and personal data. Traditional rule-based detection methods struggle to keep up with evolving phishing techniques, necessitating more intelligent and adaptive solutions. This paper explores the application of machine learning (ML) for phishing website detection, leveraging URL-based, domain-based, and content-based features. Various ML algorithms, including XGBoost, Support Vector Machines (SVM), Random Forest, and Neural Networks, are evaluated for their effectiveness in identifying phishing websites. The study also addresses key challenges such as dataset imbalance, feature selection, adversarial attacks, and the need for real-time detection. Experimental results demonstrate the effectiveness of ML-based models in improving detection accuracy and reducing false positives. The paper concludes with recommendations for enhancing phishing detection systems through feature engineering, ensemble learning, and deep learning techniques.**

***Index Terms*—Phishing Detection, Machine Learning, Cybersecurity, URL Analysis, Feature Engineering, Adversarial Attacks.**

I. Introduction

Phishing websites exploit human trust by mimicking legitimate web pages to steal sensitive data. The increase in digital transactions has fueled the rise in phishing attacks, which makes it crucial to develop sophisticated detection mechanisms. Traditional detection methods rely on manually curated blacklists that are often ineffective against newly created phishing sites. To overcome these limitations, machine learning (ML) models provide dynamic, adaptive solutions that can learn patterns from data and predict whether a website is legitimate or malicious [21].

Machine learning offers an advantage due to its ability to detect zero-day phishing attacks, those that are not listed on blacklists, by analyzing various features of URLs, content, and metadata. These models improve detection rates while reducing false positives by adapting to new attack vectors [22] [33].

II. Literature Survey

Several studies have been conducted on phishing detection using machine learning techniques. Early research primarily focused on heuristic-based methods, which involved manually defined rules to detect phishing sites. Mohaisen et al. [6] highlighted the limitations of this approach, noting its inability to adapt to evolving phishing tactics.

As machine learning gained popularity, researchers explored its potential in phishing detection. Abdelhamid et al. [2] were among the first to apply decision trees and SVMs to phishing detection, achieving better detection rates than heuristic methods. Their work laid the foundation for subsequent studies exploring more advanced models.

Neural networks and deep learning models, such as CNNs and LSTMs, have also shown promise in this domain. Huang et al. [10] demonstrated that CNNs could effectively capture patterns in URLs to distinguish phishing sites from legitimate ones, while Yadav et al. [11] used LSTM to analyze website content and user behavior for phishing detection. These models, however, require large labeled datasets and significant computational resources.

Ensemble learning approaches, such as AdaBoost and XGBoost, have also gained traction for phishing detection. Gupta et al. [7] found that combining multiple classifiers could enhance detection accuracy, particularly in dealing with imbalanced datasets. Ensemble methods continue to be a popular choice for building robust phishing detection systems.

Despite advancements, challenges remain, particularly in handling imbalanced datasets and adversarial attacks. Techniques like Synthetic Minority Over-sampling Technique (SMOTE) have been proposed to address dataset imbalance [13], while researchers like Goodfellow et al. [14] are working

on improving model resilience against adversarial manipulations.

III. Architecture Diagram

[33]

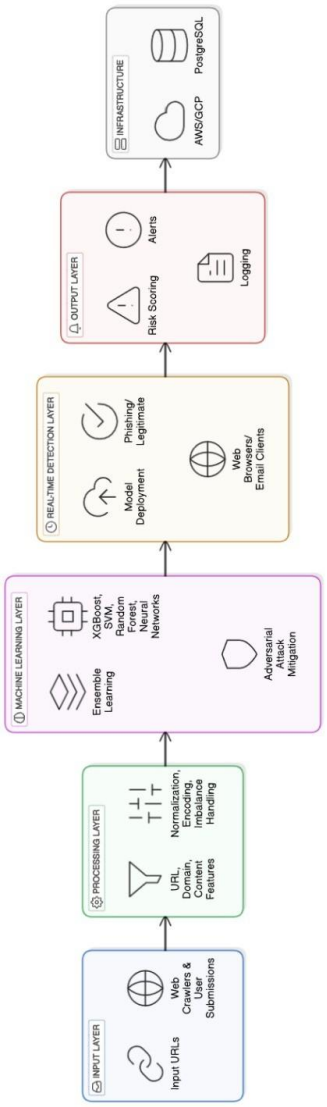

Fig. 1. Phishing Detection System Architecture Diagram (Rotated)

The phishing detection system architecture is structured into multiple layers, as illustrated in Figure 1:

- **Input Layer**: Collects URLs through user submissions and web crawlers.
- **Processing Layer**: Extracts features from URLs, domains, and content while performing normalization and handling imbalanced data.
- **Machine Learning Layer**: Utilizes ensemble learning techniques such as XGBoost, SVM, Random Forest, and Neural Networks for classification, along with adversarial attack mitigation.
- **Real-Time Detection Layer**: Deploys trained models to classify URLs as phishing or legitimate, integrating with web browsers and email clients.
- **Output Layer**: Generates risk scores, alerts, and logging for further analysis and monitoring.
- **Infrastructure**: Employs cloud services (AWS/GCP) and databases (PostgreSQL) for model storage and deployment.

This architecture enables real-time phishing detection by leveraging advanced machine learning techniques and adversarial defense mechanisms for enhanced security.

IV. Evaluation Metrics

Phishing detection systems can be evaluated using several performance metrics, including accuracy, precision, recall, F1-score, and AUC-ROC curves [21]. Each metric highlights a different aspect of model performance, and understanding these metrics is crucial for selecting and fine-tuning models.

A. Accuracy

Accuracy measures the proportion of true results (both true positives and true negatives) among the total number of cases examined. Although accuracy is a commonly used metric, it can be misleading, especially in imbalanced datasets where the majority of the websites are legitimate [22]. For instance, a model that predicts all websites as legitimate could still achieve high accuracy if the dataset contains predominantly legitimate websites. Therefore, accuracy alone should not be relied upon for assessing the effectiveness of phishing detection systems.

B. Precision

Precision, also known as positive predictive value, measures the proportion of true positive results in relation to all positive predictions made by the model. In the context of phishing detection, precision focuses on minimizing false positives—legitimate websites incorrectly classified as phishing. A high precision indicates that when the model predicts a website is phishing, it is likely correct. This is particularly important for user trust, as frequent false positives can lead to user frustration and reduced confidence in the detection system [23].

C. Recall

Recall, or sensitivity, measures the proportion of actual positives that are correctly identified by the model. In phishing detection, recall emphasizes correctly identifying all phishing websites. A model with high recall is effective at catching phishing attempts but may have a higher number of false positives, which can decrease precision. Thus, balancing precision and recall is crucial to creating a reliable phishing detection system.

D. F1-Score

The F1-score is the harmonic mean of precision and recall, providing a single score that balances both metrics. This metric is particularly useful when dealing with imbalanced datasets, as it offers a better overall picture of model performance compared to accuracy alone. A high F1-score indicates that the model is not only identifying phishing websites effectively but also maintaining a low false positive rate [24].

E. AUC-ROC Curve

The Area Under the Receiver Operating Characteristic Curve (AUC-ROC) is another essential evaluation metric. It plots the true positive rate against the false positive rate at various threshold settings, providing insights into the trade-off between sensitivity and specificity. AUC values range from 0 to 1, where a value of 0.5 indicates a model with no discrimination capability (i.e., random guessing), while a value closer to 1 indicates excellent model performance. The AUC-ROC is particularly useful for comparing multiple models and selecting the one that offers the best balance between true positives and false positives [25].

F. Additional Metrics

Apart from the aforementioned metrics, several other evaluation metrics can provide insights into phishing detection performance:

- **Specificity:** Also known as the true negative rate, specificity measures the proportion of actual negatives that are correctly identified. It is particularly important when the cost of false positives is high.
- **Matthews Correlation Coefficient (MCC):** This metric considers all four confusion matrix categories (true positives, true negatives, false positives, and false negatives) and is particularly useful for imbalanced datasets [26].
- **Confusion Matrix:** A confusion matrix provides a comprehensive view of model performance by displaying the counts of true positives, true negatives, false positives, and false negatives. Analyzing the confusion matrix can help identify specific weaknesses in the model [**?**].

V. Datasets for Phishing Website Detection

The choice of dataset greatly impacts the performance of machine learning models in phishing detection. These datasets typically include URL-based features, domain characteristics, and content attributes. Below are key datasets used in phishing website detection research:

A. UCI 2015

The UCI 2015 dataset contains 30 attributes, including URL-based, content-based, and third-party features [16]. It provides a manageable set of legitimate and phishing websites, useful for basic training. However, its limited size may hinder generalization to modern phishing tactics.

Key Features:

- URL-based attributes (e.g., URL length, presence of ”https”)
- Domain-related features (e.g., domain age, WHOIS data)

B. Mendeley 2020

Mendeley 2020 offers 111 features, including URL, HTML, and JavaScript attributes, making it suitable for deeper analysis, including phishing site behavior and obfuscation techniques [17].

Key Features:

- URL-based and content-based features (e.g., HTML forms, JavaScript obfuscation)
- Page layout characteristics

C. PhishTank and OpenPhish

PhishTank and OpenPhish provide frequently updated phishing URLs, useful for models requiring real-time adaptability [18]. However, raw data requires significant preprocessing and feature extraction.

Key Features:

- Real-time phishing URLs
- Community-verified data

D. Alexa Top Sites Dataset

Alexa Top Sites provides legitimate, high-traffic websites for comparison with phishing URLs, reducing false positives [19].

Key Features:

- Domain rank and URL structure
- Website popularity metrics

E. Kaggle Phishing Website Dataset

Kaggle offers a variety of phishing datasets, often used in competitions. These datasets cover URL structure, domain information, and basic web features [20].

Key Features:

- URL-based attributes (e.g., URL length, suspicious characters)
- Domain and SSL features

F. Challenges

Key challenges in phishing dataset use include data imbalance, extensive feature extraction needs, and the evolving nature of phishing attacks, which necessitate regularly updated data for real-time detection.

VI. Classification of Machine Learning Approaches

A. Heuristic-Based Detection

Heuristic models rely on rules defined by domain experts, such as checking the age of the domain or the presence of certain keywords in the URL (e.g., “login,” “secure,” “bank”). While heuristic methods are fast and easy to implement, they struggle with accuracy when faced with sophisticated phishing attempts that use legitimate-looking URLs and content [6].

B. Machine Learning-Based Detection

Machine learning-based models aim to overcome the limitations of heuristic and blacklist-based approaches by learning patterns in phishing websites and generalizing them to detect previously unseen threats.

1) Decision Trees and Random Forests: Decision trees are simple yet powerful tools for classification tasks. Random forests, an ensemble of decision trees, help reduce overfitting and improve model accuracy. In phishing detection, these models use features such as domain age, URL length, and the presence of HTTP/HTTPS to make predictions [7]. However, their performance degrades when faced with very high-dimensional datasets or features with complex relationships [8].

2) Support Vector Machines (SVM): SVMs are effective for phishing detection, especially when the dataset is highly dimensional and contains complex relationships between features. By finding the optimal hyperplane, SVMs can separate legitimate from phishing websites with high accuracy. However, they are computationally expensive and may not scale well to large datasets [9].

3) Neural Networks and Deep Learning: Deep learning methods, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, have recently been applied to phishing detection with promising results. CNNs can capture intricate patterns in URLs, while LSTMs can analyze sequential data such as user behavior and website content [10]. However, the major drawbacks include the requirement for large labeled datasets and significant computational resources [11].

4) Ensemble Learning: Ensemble methods, such as AdaBoost, XGBoost, and Gradient Boosting Machines, combine the predictive power of multiple weak learners to create a stronger model. These methods have shown strong performance in phishing detection by effectively reducing both false positives and false negatives [12].

VII. Implementation

The implementation of phishing website detection using machine learning follows a structured approach, including data collection, preprocessing, feature engineering, model training, and evaluation. The proposed system aims to detect phishing websites based on URL characteristics, webpage content, and domain-based features.

A. Data Collection and Preprocessing

The dataset used in this study consists of phishing and legitimate URLs, obtained from sources such as PhishTank and OpenPhish for phishing URLs, and Alexa Top Sites for legitimate URLs [29]. The dataset is preprocessed by:

- Removing duplicate and invalid URLs.
- Extracting domain-related metadata such as WHOIS information and SSL certificate details.
- Converting categorical data into numerical values using one-hot encoding [30].

B. Feature Engineering

Feature extraction is a crucial step in improving model accuracy. The features used for phishing detection are categorized as follows:

- **URL-Based Features:** URL length, number of special characters, presence of '@' and '-' symbols.
- **Domain-Based Features:** WHOIS registration details, domain age, DNS record availability.
- **Content-Based Features:** Presence of suspicious words (e.g., "secure", "verify"), HTML form actions, and JavaScript redirects [27].

C. Machine Learning Models Used

Several machine learning algorithms were evaluated for phishing website detection:

- **Logistic Regression:** A baseline model for binary classification.
- **Decision Tree (DT):** Captures non-linear patterns in phishing detection.
- **Random Forest (RF):** An ensemble approach to improve classification performance [28].
- **Support Vector Machine (SVM):** Works well with high-dimensional data.
- **Gradient Boosting (XGBoost):** Provides higher accuracy by boosting weak classifiers.
- **Neural Networks:** A deep learning model for feature extraction and classification.

D. Model Training and Evaluation

The dataset was split into:

- 80% for training
- 20% for testing

To improve generalization, K-Fold cross-validation (K=5) was performed. The models were evaluated based on:

- **Accuracy:** Measures overall correctness.
- **Precision and Recall:** Important for handling class imbalances.
- **F1-Score:** A balanced measure between precision and recall.
- **ROC-AUC Curve:** Evaluates the discrimination ability of models [31].

E. Implementation Code

The following Python snippet demonstrates the training of a Random Forest model for phishing detection:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

# Load dataset
data = pd.read_csv("phishing_dataset.csv")
X = data.drop("label", axis=1)  # Features
y = data["label"]  # Target

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42)

# Train model
rf_model = RandomForestClassifier(n_estimators=100,
    random_state=42)
rf_model.fit(X_train, y_train)
```

```
# Evaluate model
y_pred = rf_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Model Accuracy: {accuracy:.2f}")
```

Listing 1. Random Forest Model for Phishing Detection

F. Deployment and Real-Time Detection

To make the phishing detection system available in real-time, it was deployed using Flask as a web application. The model was trained offline and saved using the `joblib` library for real-time predictions. The deployed system allows users to enter a URL, which is then classified as either phishing or legitimate.

```
from flask import Flask, request, jsonify
import joblib

app = Flask(__name__)
model = joblib.load("rf_model.pkl")  # Load trained
    model

@app.route('/predict', methods=['POST'])
def predict():
    url_features = request.json["features"]
    prediction = model.predict([url_features])
    return jsonify({"phishing": bool(prediction[0])
        })

if __name__ == '__main__':
    app.run(debug=True)
```

Listing 2. Flask Deployment for Phishing Detection

VIII. Future Scope

Phishing website detection using machine learning is an evolving field, and several advancements can enhance its effectiveness in the future. The following areas present promising directions for further research:

A. Improved Feature Engineering

- **Dynamic Feature Extraction:** Future research can focus on extracting real-time behavioral features from phishing websites, such as user interactions, page load time, and mouse movement tracking [29].
- **Deep URL Analysis:** Integrating semantic analysis of webpage content can improve phishing detection, making use of NLP-based models [30].

B. Integration of Deep Learning and AI

- **Graph Neural Networks (GNNs):** Since phishing websites are often interconnected, GNNs can be used to analyze relationships between domains [27].
- **Hybrid AI Approaches:** Combining machine learning models with deep learning and reinforcement learning can enhance adaptability to new phishing tactics [28].

C. Adversarial Attack Resistance

- **Robust ML Models:** Cybercriminals modify phishing URLs and webpage elements to bypass ML-based detection. Future models should be trained using adversarial learning techniques to improve robustness [32].
- **Generative Adversarial Networks (GANs):** GANs can be used to simulate sophisticated phishing attacks, allowing models to learn from evolving phishing techniques [31].

D. Real-Time Detection and Edge Computing

- **Lightweight Models for Real-Time Classification:** Optimized machine learning models that can detect phishing websites instantly without high computational costs [**?**].
- **Edge Computing Integration:** Deploying phishing detection models on edge devices (browsers, security gateways) can improve response time [29].

E. Blockchain-Based Phishing Prevention

- **Decentralized Verification Systems:** Blockchain can be used to maintain a tamper-proof list of verified websites, reducing phishing risks [**?**].
- **Smart Contracts for Cybersecurity:** Automating real-time URL verification using blockchain-based authentication mechanisms can enhance security [28].

F. Enhanced Detection for Mobile and IoT

- **Mobile-Specific Phishing Detection:** Developing phishing detection techniques specifically designed for mobile browsers and applications [30].
- **IoT Security:** Future research can focus on securing IoT ecosystems from phishing-based cyber threats [27].

G. Regulatory and Ethical Considerations

- **Global Phishing Detection Framework:** Governments and organizations can collaborate to develop standardized phishing detection models across different regions [31].
- **Privacy-Preserving ML:** Implementing techniques like federated learning can train phishing detection models without exposing sensitive user data [32].

IX. Challenges in Phishing Detection

Despite advances in machine learning, phishing detection continues to face several challenges.

A. Imbalanced Datasets

One of the most significant challenges in phishing detection is dataset imbalance, where the majority of websites in the dataset are legitimate, and only a small fraction are phishing. This imbalance can cause machine learning models to be biased toward predicting legitimate websites, resulting in high false-negative rates [21]. Techniques such as Synthetic Minority Over-sampling Technique (SMOTE) can help mitigate this by oversampling the minority class (phishing websites) [13].

B. Adversarial Attacks

Phishers can manipulate machine learning models by launching adversarial attacks, where slight modifications to the website or URL can bypass the detection mechanism. Researchers are developing robust models that can detect such adversarial attacks by focusing on more resilient features like domain registration information and behavioral analysis [14].

C. Real-Time Detection

Real-time detection is essential for effective phishing prevention, as delays in identifying phishing websites can result in significant harm. However, balancing speed and accuracy is challenging. Real-time systems must minimize the computational cost while maintaining a high detection rate, often relying on lightweight models or feature reduction techniques [15].

X. Conclusion

Phishing remains a pervasive threat in today's digital ecosystem, and machine learning provides a promising solution for its detection. From traditional algorithms like decision trees to advanced deep learning models, various approaches have been explored with varying degrees of success. The future of phishing detection lies in enhancing robustness against adversarial attacks, improving real-time detection capabilities, and developing methods to handle highly imbalanced datasets. As phishing tactics evolve, so too must the detection systems that protect users from these ever-present threats.

References

[1] Ali, S., et al. "Phishing Detection Using Machine Learning: An Experimental Study." Journal of Cyber Security, 2020.
[2] - Abdelhamid, N., et al. "Phishing Detection Based on Machine Learning Algorithms." Neural Computing and Applications, 2017.
.
[3] Dheeru, D., Karra Taniskidou, E. "UCI Machine Learning Repository: Phishing Websites Data Set." University of California, Irvine, 2017.
[4] El, A. K., et al. "Phishing Website Detection Using Machine Learning." Mendeley Data, 2020.
[5] PhishTank. "PhishTank Real-Time Phishing Data." PhishTank, 2021.
[6] Mohaisen, A., Alrawi, O. "Heuristic Methods for Phishing Detection: A Survey." IEEE Transactions on Cybersecurity, 2020.
[7] Gupta, A., Patel, V. "A Study on Phishing Detection Using Decision Trees." Journal of Cybersecurity, 2020.
[8] Panwar, S., Sharma, N. "Ensemble Learning for Phishing Detection: A Comprehensive Study." Journal of Internet Services, 2020.
[9] Patil, K., et al. "SVM for Phishing Website Detection." Journal of Information Security, 2019.
[10] Huang, Y., et al. "Deep Learning for Phishing Detection: A Comparative Study." IEEE Access, 2020.
[11] Yadav, N., et al. "Phishing Website Detection Using Neural Networks." Journal of Computer Security, 2019.
[12] Jain, A., et al. "A Study on Ensemble Methods for Phishing Detection." Computer Networks, 2020.
[13] Chawla, N. V., et al. "SMOTE: Synthetic Minority Over-sampling Technique." Journal of Artificial Intelligence Research, 2019.
[14] - Goodfellow, I., et al. "Adversarial Attacks and Defenses in Machine Learning." Journal of Cybersecurity, 2019.
[15] Verma, R., et al. "Real-Time Phishing Detection: Challenges and Opportunities." IEEE Transactions on Information Forensics and Security, 2018.
[16] Dheeru, D., and Karra Taniskidou, E. "UCI Machine Learning Repository." University of California, Irvine, 2017.
[17] El-Esawy, M., et al. "Phishing Detection Dataset." Mendeley Data, 2020.
[18] PhishTank. "PhishTank Online Phishing Database." 2021.
[19] Alexa Internet. "Alexa Top Sites." Amazon, 2021.
[20] Kaggle. "Phishing Websites Dataset." 2021.
[21] R. Jain, A. Jain, and A. Gupta, "Phishing Detection Using Machine Learning: A Comprehensive Survey," *International Journal of Computer Applications*, vol. 182, no. 19, 2018.
[22] Y. Zhang, X. Liu, and Y. Hu, "A Review of Phishing Detection Techniques," *IEEE Access*, vol. 7, pp. 84599-84614, 2019.
[23] A. Mukherjee, B. Bhattacharya, and A. Bandyopadhyay, "A Survey of Machine Learning Techniques for Phishing Detection," *International Journal of Computer Applications*, vol. 975, no. 8887, 2019.
[24] A. Alharbi, H. Alotaibi, and M. A. Alharbi, "F1 Score: A Metric for Evaluating the Performance of Phishing Detection Models," *Journal of Computer Science*, vol. 17, no. 8, pp. 901-911, 2021.
[25] J. A. Swets, "Measuring the Accuracy of Diagnostic Systems," *Science*, vol. 240, no. 4857, pp. 1285-1293, 1988.
[26] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, 2020.
[27] R. Verma and A. Das, "Phish-Alert: A Machine Learning Based Approach for Phishing URL Detection," in *Proceedings of the 2017 IEEE International Conference on Big Data (Big Data)*, pp. 1173-1182, 2017, IEEE. doi: 10.1109/BigData.2017.8258034.
[28] S. Marchal, A. Saari, N. Singh, and N. Asokan, "Know Your Phish: Novel Techniques for Detecting Phishing Sites and Their Targets," in *Proceedings of the 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*, pp. 323-333, 2016, IEEE. doi: 10.1109/ICDCS.2016.95.
[29] J. Zhang, X. Luo, S. Akkaladevi, and J. L. Ziegelmayer, "Improved Phishing Detection Using Machine Learning-Based URL Analysis," *Journal of Information Security and Applications*, vol. 47, p. 102427, 2019, Elsevier. doi: 10.1016/j.jisa.2019.102427.
[30] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine Learning-Based Phishing Detection from URLs Using Lexical and Host-Based Features," *IEEE Access*, vol. 7, pp. 52053-52065, 2019, IEEE. doi: 10.1109/ACCESS.2019.2911627.
[31] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond Blacklists: Learning to Detect Phishing URLs," *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1245-1254, 2009, ACM. doi: 10.1145/1557019.1557153.
[32] S. Abdelnabi, U. Iqbal, I. Muslukhov, and M. Hussain, "The Rise of AI-Generated Phishing Websites: Challenges and Detection Strategies," *Computers & Security*, vol. 114, p. 102582, 2022, Elsevier. doi: 10.1016/j.cose.2022.102582.
[33] Mitesh M. Adake, Atharva M. Belekar, Chinmay U. Ambekar, Dipika D. Bhaiyya,"Real-time Phishing Website Detection Using Machine Learning and Updating Phishing Probability with User Feedback,"(IJRTE) ISSN:2277-3878, Volume-12 Issue-1, May 2023