

# Employing Gradient Boosting-Steepest Descent Approach for Identifying Potential Phishing Attacks

Narendra Kumar Bairagi<sup>1</sup>, Prof. Vipin Kasera<sup>2</sup>  
Research Scholar<sup>1</sup>, Asst. Professor<sup>2</sup>  
VITM, Indore, India<sup>1,2</sup>

**Abstract:** *Phishing attacks are among the most common cyber threats, tricking users into revealing sensitive information such as passwords, credit card details, and personal data. Traditional security measures, such as rule-based detection and blacklists, struggle to keep up with the evolving tactics of cybercriminals. Phishing attacks typically occur through emails, fake websites, or fraudulent messages that appear legitimate. Attackers use social engineering techniques to manipulate users into clicking malicious links or downloading harmful attachments. Due to the dynamic nature of phishing strategies, static detection methods often fail to provide adequate protection. Advanced machine learning techniques, particularly deep learning, have proven to be more effective in identifying complex patterns in phishing attempts. This paper presents a Gradient Boosting-Steepest Descent Approach for identifying potential phishing attacks. It has been shown that the proposed approach outperforms existing baseline models in terms of classification accuracy.*

**Keywords:** *Cyber Security, Phishing Attacks, Gradient Boosting, Steepest Descent, Deep Neural Networks, Classification Accuracy.*

## I. INRRODUCTION

Phishing attacks are a major cybersecurity threat, tricking individuals and organizations into revealing sensitive information such as login credentials, financial details, and personal data. Cybercriminals use deceptive tactics to impersonate trusted entities and manipulate victims into taking harmful actions. Various types of phishing attacks exist, each with distinct characteristics and targets. To combat these threats, organizations and individuals must implement effective countermeasures. The most common types of

phishing attacks and their common countermeasures are presented next 1]:

**1. Email Phishing:** Email phishing is the most common type of phishing attack, where attackers send fraudulent emails that appear to be from legitimate sources. These emails often contain malicious links, fake login pages, or infected attachments designed to steal user credentials or install malware. Attackers use urgency, fear, or enticing offers to trick recipients into responding [2]

**Countermeasures:** To prevent email phishing, organizations should implement email filtering systems, such as spam filters and machine learning-based anomaly detection. Users should verify the sender's email address, avoid clicking on suspicious links, and enable multi-factor authentication (MFA) to add an extra layer of security. Regular phishing awareness training can also help individuals recognize and avoid phishing attempts.

**2. Spear Phishing:** Spear phishing is a targeted attack where cybercriminals tailor phishing emails to a specific individual or organization. Unlike generic email phishing, spear phishing involves extensive research on the target, making the attack more convincing. Attackers may impersonate a colleague, boss, or business partner to gain trust and trick the victim into revealing sensitive information [3].

**Countermeasures:** Defending against spear phishing requires advanced email security solutions that detect anomalies in communication patterns. Employees should be trained to verify requests for sensitive information, especially those involving financial transactions. Implementing digital signatures and secure email gateways can help authenticate legitimate communications and prevent impersonation attacks.

**3. Whaling:** Whaling is a form of spear phishing that targets high-profile individuals such as executives, CEOs, and government officials. Attackers craft highly sophisticated emails that appear to be urgent business requests, legal matters, or financial transactions. The goal is to manipulate senior officials into authorizing fraudulent wire transfers or disclosing confidential company data [4].

**Countermeasures:** Organizations should establish strict verification protocols for high-value transactions, such as requiring multiple approvals for financial transfers. Educating executives about phishing risks and implementing advanced threat protection tools can help detect and prevent whaling attacks. Using encrypted communication channels for sensitive discussions also reduces the risk of information exposure.

**4. Smishing (SMS Phishing):** Smishing involves sending fraudulent messages via SMS or messaging apps to deceive victims into clicking malicious links or providing personal information. These messages often claim to be from banks, government agencies, or delivery services, urging recipients to take immediate action.

**Countermeasures:** To counter smishing, users should avoid clicking on links in unsolicited messages and verify the authenticity of messages by directly contacting the organization. Mobile security applications and SMS filtering tools can help block suspicious messages. Telecom providers can also implement fraud detection mechanisms to reduce smishing incidents.

**5. Vishing (Voice Phishing):** Vishing is a phishing attack carried out over phone calls, where scammers pose as legitimate representatives from banks, tech support, or government agencies. Attackers use social engineering tactics to persuade victims into revealing sensitive information or making fraudulent payments [5].

**Countermeasures:** Users should be cautious when receiving unexpected calls requesting sensitive information. Banks and organizations should educate customers about their official communication policies, emphasizing that they will never ask for personal

details over the phone. Implementing call authentication and blocking suspicious numbers can help mitigate vishing attacks.

**6. Clone Phishing:** Clone phishing involves replicating a legitimate email and modifying its content to include malicious links or attachments. Attackers send the cloned email from a spoofed address that appears to be from a trusted sender, tricking recipients into clicking harmful links [6].

**Countermeasures:** To prevent clone phishing, organizations should implement email authentication protocols such as DMARC (Domain-based Message Authentication, Reporting, and Conformance), SPF (Sender Policy Framework), and DKIM (DomainKeys Identified Mail). Employees should verify any unexpected attachments or links, even if the email appears familiar.

**7. Website Spoofing and Pharming:** In website spoofing, attackers create fake websites that mimic legitimate ones to steal login credentials. Pharming is a related attack where cybercriminals manipulate DNS settings to redirect users to fraudulent websites without their knowledge [7].

**Countermeasures:** Users should always check the website URL and look for security indicators such as HTTPS and SSL certificates before entering sensitive information. Organizations can use anti-phishing browser extensions and domain monitoring tools to detect and block spoofed websites. Deploying DNS security solutions helps prevent unauthorized domain redirection [8].

## II. MACHINE LEARNING MODELS FOR IDENTIFYING PHISHING ATTACKS:

Traditional security measures often fail to detect sophisticated phishing attempts, making machine learning (ML) an essential tool for improving detection. ML models analyze patterns in phishing emails, URLs, and website structures to identify malicious content effectively. This section presents the

most common machine learning and deep learning models for identifying phishing attacks [9]:

## 2.1 Machine Learning Models:

### 1. Decision Trees

Decision trees classify phishing and legitimate instances based on a sequence of decisions derived from specific features such as URL length, domain age, and the presence of suspicious keywords. Each node in the tree represents a decision rule, and the model navigates through these rules to reach a final classification [10].

#### *Advantages*

Easy to interpret and implement  
Works well with structured data  
Requires minimal preprocessing

#### *Limitations*

Prone to overfitting, leading to poor generalization  
Less effective for complex, evolving phishing patterns

### 2. Random Forest

Random forest is an ensemble learning method that combines multiple decision trees to enhance classification accuracy. By aggregating the outputs of several trees, it reduces overfitting and improves robustness.

#### *Advantages*

More accurate than a single decision tree  
Handles large datasets effectively  
Resistant to noise and outliers

#### *Limitations*

Computationally expensive  
Less interpretable compared to individual decision trees [11].

### 3. Support Vector Machines (SVM)

SVM is a powerful classification model that finds the optimal hyperplane to separate phishing and legitimate instances based on extracted features. It is particularly effective in high-dimensional spaces.

#### *Advantages*

High accuracy in detecting phishing attempts  
Works well with small datasets

#### *Limitations*

Computationally intensive for large datasets  
Requires careful feature selection and tuning [12]

### 4. Naïve Bayes

Naïve Bayes is a probabilistic classification model based on Bayes' theorem. It is widely used in email phishing detection, where it classifies emails as phishing or legitimate based on word frequency and patterns [13].

#### *Advantages*

Fast and computationally efficient  
Works well with text-based classification tasks

#### *Limitations*

Assumes independence among features, which may not always be valid  
Less effective against sophisticated phishing tactics

### 5. Logistic Regression

Logistic regression is a simple yet effective ML model that calculates the probability of a given instance being phishing or legitimate based on a weighted sum of its features [14].

#### *Advantages*

Easy to interpret and implement  
Works well for simple phishing detection tasks

#### *Limitations*

Struggles with complex and nonlinear patterns  
Requires well-engineered features for optimal performance

### 6. K-Nearest Neighbors (KNN)

KNN classifies a phishing attempt by comparing it with the closest labeled data points. It calculates the distance between a new instance and its nearest neighbors in the dataset [15].

#### *Advantages*

Simple to implement  
Effective for small datasets

#### *Limitations*

Inefficient for large datasets  
Sensitive to irrelevant features

## 2.2 Deep Learning Models:

### 1. Artificial Neural Networks (ANNs)

ANNs consist of multiple interconnected layers of neurons that process information through weighted connections. They are used to classify phishing attacks based on various features extracted from emails, URLs, and website content [16]

#### *Advantages*

Can learn complex patterns in phishing attempts  
Adaptable to different data types (text, URLs, images)

#### **Limitations**

Requires large datasets for training  
Computationally expensive

## **2. Convolutional Neural Networks (CNNs)**

CNNs are commonly used in image and text processing tasks. In phishing detection, they analyze website screenshots, email structures, and other visual cues to identify fraudulent activity [17].

#### **Advantages**

Effective in detecting phishing through image and text analysis  
Automatically extracts features without manual intervention

#### **Limitations**

Requires high computational power  
Needs large labeled datasets

## **3. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM)**

RNNs and LSTMs are designed for sequential data analysis, making them ideal for analyzing email content and URL sequences. They remember past information, allowing them to detect patterns in phishing emails and malicious URLs [18]

#### **Advantages**

Effective in analyzing email sequences and phishing content  
Can capture temporal dependencies in phishing patterns

#### **Limitations**

Training can be slow and resource-intensive  
Prone to vanishing gradient issues in long sequences

## **4. Transformer-Based Models (BERT, GPT)**

Transformer-based models like BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) are revolutionizing phishing detection. They use deep contextual understanding to detect phishing attempts in emails and messages [19].

#### **Advantages**

Highly accurate in understanding phishing content  
Can process large amounts of text data efficiently

#### **Limitations**

Requires significant computational resources

Needs large, well-labeled datasets

## **III. PROPOSED METHDOLOGY**

The proposed methodology presents a Gradient Boosting-Steepest Descent approach for identifying phishing attacks through relevant web features.

The essence of the approach is leveraging the advantages of gradient boosting, while limiting the computational complexity through the steepest descent approach.

### **3.1 Gradient Boosting (XG-Boost):**

The gradient boosting (XG-Boost) is a powerful technique used to improve the performance of predictive models by sequentially combining weak learners into a strong ensemble. Gradient boosting is an iterative ensemble learning technique that builds models sequentially, with each new model correcting the errors of the previous ones. Unlike bagging (e.g., random forests), which trains multiple models independently, boosting focuses on improving weak models by optimizing their weights using gradient descent. In gradient boosting, a loss function measures the model's performance, and gradients guide the adjustments needed to minimize this loss. The process involves three main components [20]:

1. Loss function – Measures the error between predictions and actual values.
2. Weak learners – Typically decision trees or shallow neural networks trained to correct previous errors.
3. Additive model – Combines weak learners iteratively to reduce the overall error.

### **3.2 Gradient Boosting for Deep Nets:**

Gradient boosting can use shallow neural networks as weak learners. Each neural network learns to correct the errors of the previous ones, gradually improving overall performance. This approach is useful in situations where traditional DNNs struggle with training efficiency or require additional bias correction.

This approach has the following distinctive advantages:



Gradient boosting helps correct errors progressively, leading to more accurate predictions compared to standalone deep neural networks.

Since boosting assigns more weight to difficult samples, it can prevent the deep network from overfitting to easy patterns. Additionally, regularization techniques such as shrinkage (learning rate control) and subsampling help improve generalization.

By focusing on the hardest-to-learn samples at each iteration, gradient boosting can speed up training convergence, reducing the number of epochs required for effective learning [21].

### 3.3 Steepest Descent Approach:

The steepest descent approach has been used to train the XG-Boost based Deep Neural Network model so as to limit the computational complexity. Optimization algorithms play a crucial role in training machine learning and deep learning models by minimizing loss functions and improving model performance. Among various optimization methods, Steepest Descent and Scaled Conjugate Gradient (SCG) are widely used to efficiently find optimal solutions in high-dimensional spaces. Steepest Descent is a fundamental gradient-based approach, while SCG improves convergence speed by addressing the limitations of traditional gradient methods. The weight update rule for conventional gradient descent is [22]:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \alpha \nabla f(\mathbf{w}_k, \mathbf{b}) \quad (1)$$

The weight update rule for the modifies scaled conjugate gradient descent is:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - p_k \alpha \nabla f(\mathbf{w}_k, \mathbf{b}) \quad (2)$$

Here,

$\mathbf{w}_{k+1}$  denotes the weight corresponding to iteration ' $k + 1$ '.

$\mathbf{w}_k$  denotes the weight corresponding to iteration ' $k$ '.

$\alpha$  denotes the learning rate.

$\mathbf{w}_k$  and  $\mathbf{b}$  denote the weights and biases.

$p_k$  denotes the conjugate gradient direction for iteration  $k$  which optimizes the gradient search.

$\nabla f(\mathbf{w}_k, \mathbf{b})$  denotes the gradient of the loss function.

While Steepest Descent is simple and widely used, it suffers from slow convergence and learning rate sensitivity. In contrast, SCG enhances optimization efficiency by using conjugate directions and adaptive step sizes, making it more suitable for deep learning and large-scale problems.

For gradient boosting, each weak neural network model  $f_k(x)$  is trained on the negative gradient of the residuals from the previous stage as:

$$f_{k+1}(x) = -\nabla L(y_k, F_k(x)) \quad (3)$$

And,

$$F_{k+1}(x) = F_k(x) + \alpha f_k(x) \quad (4)$$

Here,

$f_k(x)$  denotes each of the weak neural network models.

$F_k(x)$  denotes the ensemble prediction at iteration.

$\alpha$  denotes the ensemble step size or learning rate to minimize the **ensemble loss function**:

$$L = \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^N L(y_i, F_{k-1}(x_i) + \alpha f_k(x_i)) \quad (5)$$

The model is trained till convergence is reached. Typically the least squares optimization is used for the model.

## IV. EXPERIMENTAL RESULTS

The simulations are performed in MATLAB using the Deep Learning Library (ToolBox). The training to testing ratio has been take as 75:25 for 4000 samples of training and 1000 samples of testing. The results are presented next:

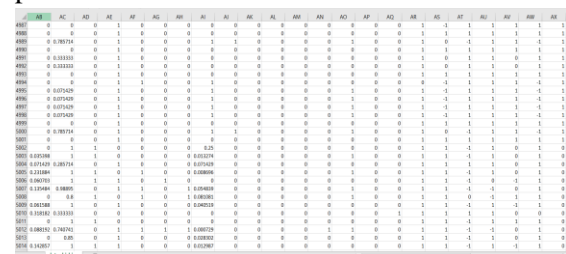


Fig.1 Raw Data

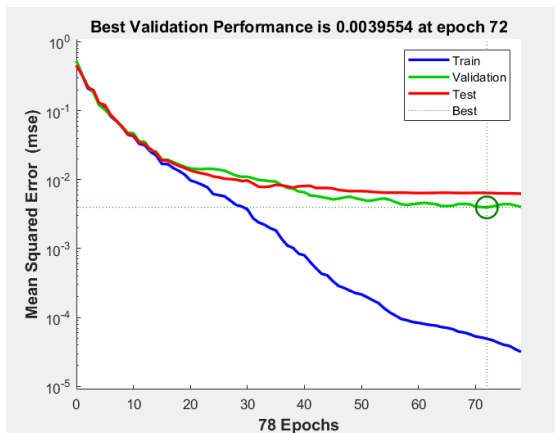


Fig.3 Iterations to convergence for overall model.

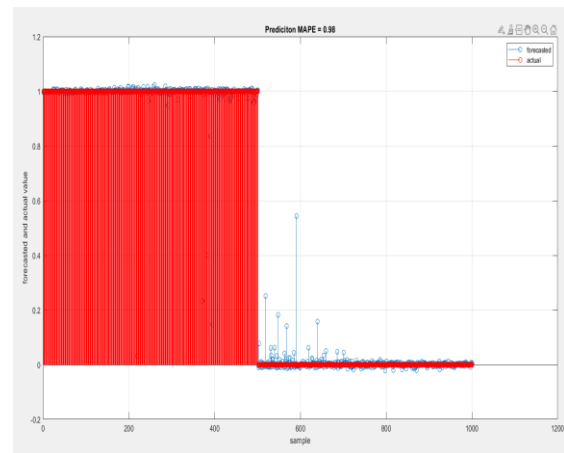


Fig.6 Error performance of overall model.

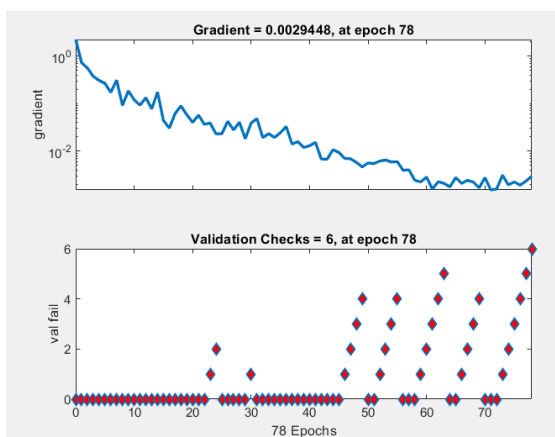


Fig.4 Gradient and validation check for overall model.

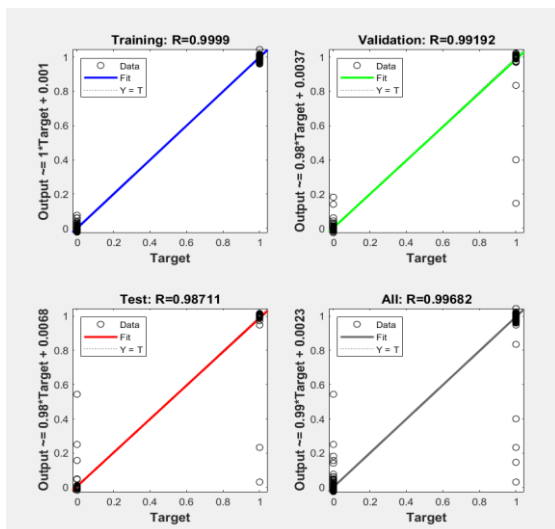


Fig.5 Regression Analysis for Overall Model

The error percentage for the model is 0.98% rendering an accuracy of 99.02%, which clearly outperforms existing work in the domain of research [23] which has a best case accuracy of 97.7% for all the models used which are:

Support Vector Machine (SVM), Boosted Decision Trees (DT), Logistic Regression (LR), Averaged Perceptron, Neural Networks, Decision Forests.

## CONCLUSION

It can be concluded that gradient boosting (XG-Boost) in deep neural network models detect phishing attacks and malware more effectively than traditional classifiers. Gradient boosting is a powerful technique that enhances deep neural network performance by iteratively refining weak learners and optimizing predictions. Gradient boosting can be applied to neural networks to improve training efficiency, reduce overfitting, and enhance accuracy. This paper presents a steepest descent approach for gradient boosting so as to classify phishing and non-phishing cases. It has been shown that the classification accuracy of the proposed model is higher compared to exiting baseline models in the domain.

## References

1. M. N. Alam, D. Sarma, F. F. Lima, I. Saha, R. -E. -, Ulfath and S. Hossain, "Phishing Attacks Detection using Machine Learning Approach," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2020, pp. 1173-1179.
2. S. Y. Yerima and M. K. Alzaylaee, "High Accuracy Phishing Detection Based on Convolutional Neural Networks," arXiv preprint arXiv:2004.03960, Apr. 2020.
3. F. Salahdine, Z. El Mrabet, and N. Kaabouch, "Phishing Attacks Detection—A Machine Learning-Based Approach," arXiv preprint arXiv:2201.10752, Jan. 2022.
4. S. Paliath, M. A. Qbeitah, and M. Aldwairi, "PhishOut: Effective Phishing Detection Using Selected Features," arXiv preprint arXiv:2004.09789, Apr. 2020.
5. P. Yang, G. Zhao and P. Zeng, "Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning," in IEEE Access 2019, vol. 7, pp. 15196-15209.
6. A. K. Jain and B. B. Gupta, "Phishing Detection: Analysis of Visual Similarity-Based Approaches," arXiv preprint arXiv:1808.08513, Aug. 2018.
7. M. A. Alauthman, A. A. Taqa, and A. A. Obaid, "Phishing Website Detection Based on Machine Learning and Feature Selection Methods," arXiv preprint arXiv:1903.12395, Mar. 2019.
8. S. Marchal, J. Francois, R. State, and T. Engel, "PhishStorm: Detecting Phishing With Streaming Analytics," IEEE Transactions on Network and Service Management, vol. 11, no. 4, pp. 458–471, Dec. 2017..
9. A. Aleroud and L. Zhou, "Phishing Environments, Techniques, and Countermeasures: A Survey," Computers & Security, vol. 68, pp. 160–196, Jul. 2017.
10. ElMouatez Billah Karbab, Mourad Debbab, Abdelouahid Derhab, Djedjiga Mouheb, "MalDozer: Automatic framework for android malware detection using deep learning", ELSEVIER 2021.
11. Shifu Hou, Aaron Saas, Lingwei Chen, Yanfang Ye, Thirimachos Bourlai, "Deep Neural Networks for Automatic Android Malware Detection", ACM 2020.
12. . Korkmaz, O. K. Sahingoz and B. Diri, "Detection of Phishing Websites by Using Machine Learning-Based URL Analysis," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-7
13. R. Vinayakumar, K. P. Soman, Prabakaran Poornachandran, "Deep android malware detection and classification", IEEE 2019.
14. Sergei Bezobrazov, Anatoly Sachenko, Myroslav Komar, Vladimir Rubanau, "The methods of artificial intelligence for malicious applications detection in android OS", IJCA 2018
15. Konstantinos Demertzis and Lazaros Iliadis, "Bio-inspired Hybrid Intelligent Method for Detecting Android Malware", Springer 2017
16. M. Khonji, Y. Iraqi, and A. Jones, "Phishing Detection: A Literature Survey," IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091–2121, Fourth Quarter 2013.
17. M. Chatterjee and A. -S. Namin, "Detecting Phishing Websites through Deep Reinforcement Learning," 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), Milwaukee, WI, USA, 2019, pp. 227-232.
18. I. Saha, D. Sarma, R. J. Chakma, M. N. Alam, A. Sultana and S. Hossain, "Phishing Attacks Detection using Deep Learning Approach," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2020, pp. 1180-1185.
19. N. Q. Do, A. Selamat, O. Krejcar, E. Herrera-Viedma and H. Fujita, "Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions," in IEEE Access, 2022, vol. 10, pp. 36429-36463.
20. S. Emami and G. Martínez-Muñoz, "Sequential Training of Neural Networks With Gradient Boosting," in IEEE Access, vol. 11, pp. 42738-42750, 2023.

21. M. Dong, L. Yao, X. Wang, B. Benatallah, S. Zhang and Q. Z. Sheng, "Gradient Boosted Neural Decision Forest," in IEEE Transactions on Services Computing, vol. 16, no. 1, pp. 330-342, 1 Jan.-Feb. 2023.
22. N. A. M. Ghani, L. C. Yeun, N. b. M. Rozar, S. b. A. Kamaruddin, S. Ibrahim and H. b. A. Rahim, "Artificial Neural Network Analysis on BMS dataset using Scale Conjugate and Gradient Descent Optimization Techniques," 2020 IEEE Conference on Big Data and Analytics (ICBDA), Kota Kinabalu, Malaysia, 2020, pp. 51-56.
23. A. Mughaid, S. AlZu'bi, A. Hnaif, S. Taamneh, A. Alnajjar "An intelligent cyber security phishing detection system using deep learning techniques", Springer 2022..