International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 7, 2025
https://www.theaspd.com/ijes.php

# Detecting Phishing Attacks Through URL Feature Analysis And Ensemble Learning

**Anusha Prakash Hippargi[1], Aruna M[2], Adarshana S[3], Priyanka D L[4], Ashish Kumar Verma[5], Samitha Khaiyum[6], Raksha Kodnad R[7]**
[1]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, anushahippargi@gmail.com
[2]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, arunam7619@gmail.com
[3]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, adarshanas129@gmail.com
[4]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, priyankadl053@gmail.com
[5]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, Kr.ashishkumarverma@gmail.com
[6]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, Samitha-mcavtu@dayanandasagar.edu
[7]Department of MCA, Dayananda Sagar College of Engineering, Bangalore, raksha-mcavtu@dayanandasagar.edu

***ABSTRACT***

*Phishing is a common cybercrime technique that takes advantage of users exposing sensitive data by forging legitimate websites. Due to attackers continuously evolving their methods, traditional rule-based detection methods cannot cope. To address this, this research explores the use of machine learning algorithms in detecting phishing URLs. The study utilizes a dataset of 11,054 URLs with 30 features derived that identify whether a website is genuine or phishing. Various machine learning models including Logistic Regression, k-Nearest Neighbors, Support Vector Machines, Decision Trees, and ensemble methods such as Random Forest, Gradient Boosting, and CatBoost were trained and evaluated.*

*Advanced data preprocessing, data exploration, and feature correlation methods were utilized for enhancing model performance. Models were evaluated based on performance measures such as accuracy, F1-score, recall, and precision. The results show that ensemble models like Random Forest and Gradient Boosting performed the highest accuracy (greater than 97%) in phishing URL classification. The results indicate that a machine learning-based solution, using various behavioral and structural URL features, can offer an effective solution to phishing detection independently and improve cybersecurity defenses.*

## 1.INTRODUCTION

Phishing has become one of the most prevalent cybercrimes in the age of the internet. Phishers create imitation websites that mimic legitimate institutions and trick individuals into sharing sensitive information such as usernames, passwords, and financial data [4]. Not only is this attack very expensive to individuals and institutions, but it also destroys faith in internet services. The fluidity of new phishing techniques—by employing techniques such as URL obfuscation, fast-flux hosting, and social engineering—poses challenges in detecting phishing sites and hence renders it a challenging research area [2].

Conventional detection techniques have employed blacklists, heuristic rules, and hand-written signatures. While these are effective against known phishing attacks, they are ineffective against new or dynamically changing threats [4]. Blacklists are regularly updated and can be the source of new attacks, generating protection gaps. Heuristic-based approaches also depend upon static patterns that may be bypassed by attackers by performing slight modifications to URLs or page content [3]. More sophisticated and dynamic approaches are therefore required in order to improve detection against.

Machine learning has also been a promising solution for detecting phishing since it can learn complex patterns from instances and apply them to new cases [1]. Supervised machine learning can classify sites as phishing or non-phishing with high accuracy by feature extraction from URLs, such as length, presence of malicious characters, or domain characteristics [2]. Ensemble techniques and advanced classifiers have been demonstrated in earlier research to outperform traditional approaches, offering a scalable and automatic solution for the real-time detection of malicious sites [1][5].

This study proposes a comprehensive study of phishing URL detection machine learning algorithms using

over 11,000 web records with 30 features. They include Logistic Regression, k-Nearest Neighbors, Support Vector Machines, Decision Trees, Random Forest, Gradient Boosting, CatBoost, and Naïve Bayes classifiers, which were compared and tested. The aim is to assess their performance in detecting phishing URLs and identify models that offer maximum accuracy and stability and hence assist in developing more efficient security systems against phishing.

## 2. LITERATURE REVIEW

### 2.1 Early Detection Strategies

Early phishing detection systems relied primarily on blacklists and heuristic rules for filtering out well-known malicious URLs. Blacklists were most commonly used by browsers and antivirus software to refuse users access to well-known phishing sites [2]. The method does not detect zero-day phishing attacks, where malicious sites are observed prior to reporting [4]. In response to this, heuristic-based systems were used, employing manually crafted rules to identify suspicious URL patterns such as unusual length, excessive subdomains, or utilization of IP addresses instead of domain names [2][3]. Although more adaptive than blacklists, these methods exhibited high false-positive rates and quickly became outdated as attackers modified their approach [7].

### 2.2 Machine Learning Techniques for Phishing Detection

Having realized the shortcomings of static methods, scientists started using supervised machine learning models for detecting phishing. Ma et al. [3] proved that a mix of lexical and host-based features could contribute significantly to the classification accuracy using machine learning models like Logistic Regression, SVM, and Naïve Bayes. Subsequent works extended this with more sophisticated feature extraction and tuning techniques to better generalize [4][5]. For example, Sahingoz et al. [4] employed a set of more than 17 URL-based features and trained several models that achieved more than 95% accuracy. These data-driven models proved effective at identifying unknown phishing attacks by learning from past data and using that knowledge on new cases.

### 2.3 Ensemble Learning Methods

Ensemble learning has been found to be a useful technique in phishing detection where multiple classifiers' outputs are merged to create robust and accurate predictions. Le et al. [10] have suggested an ensemble method using Decision Tree and SVM that outperformed single classifiers in precision and recall. Random Forest and Gradient Boosting are two models that have gained popularity as they can learn non-linear patterns and deal with noisy data [5][6]. Some studies have determined that ensemble classifiers perform better than single models on phishing datasets consistently [4][10]. Hybrid models using URL-based, content-based, and behavioral features are also being researched to improve detection performance in more sophisticated cases [9][13].

### 2.4 Feature Engineering and Performance Evaluation

Feature selection and evaluation metrics are critical components in the design of effective phishing detection systems. Researchers have been interested in extracting lexical and structural features from URLs, i.e., special character presence, HTTPS indicator, URL depth, and rare patterns in domains [8][3]. Verma and Das [8], for example, proposed efficient feature extraction techniques that removed high processing time without compromising precision. In addition to efficient feature engineering, the application of several metrics when evaluating the model, including accuracy, precision, recall, and F1-score, is critical for comprehensive evaluation [4]. Model interpretability has been a research area, including feature importance analysis as a critical component in understanding how classifiers make predictions [5][6]. Such understanding is particularly critical in cybersecurity environments where transparency and trust are crucial.

## 3. METHODOLOGY

### 3.1 Dataset Description

The data used in this research were obtained from an open-source repository with labeled examples of phishing URLs and normal URLs [1]. The data consist of 11,054 records with each record being defined in terms of 30 features that are lexical, structural, and behavioral attributes of the URLs. The target feature is a binary label of 1 or -1 to denote a phishing site or a normal site, respectively. Preprocessing of all features was performed as numerical values, enabling direct input into machine learning models without further encoding steps.

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 7, 2025
https://www.theaspd.com/ijes.php

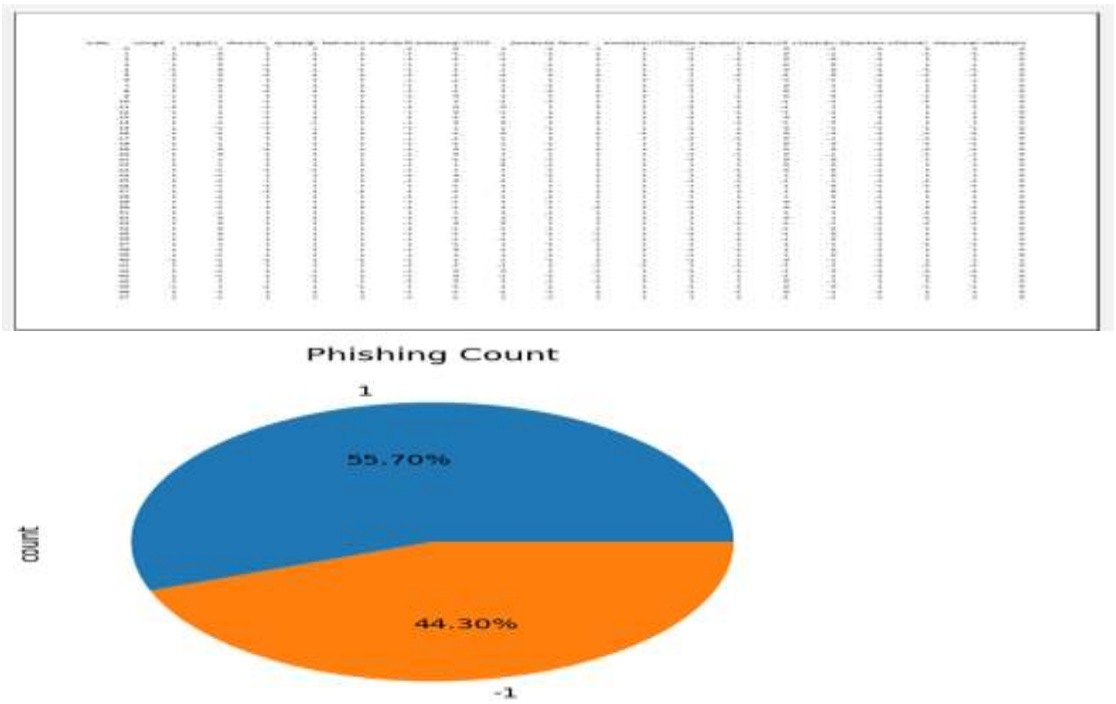


**Figure 1.** *Class distribution pie chart showing the proportion of phishing and legitimate URLs.*

### 3.2 Data Preprocessing and Exploratory Data Analysis

Preprocessing began with the elimination of unnecessary columns and verification of data integrity. Exploratory Data Analysis (EDA) was conducted to understand the distributions, identify outliers, and examine feature correlations [2]. Descriptive statistics for all features were computed. A correlation heatmap was drawn to visualize the correlations between features and cross-verify for any multicollinearity that might influence model performance [3].

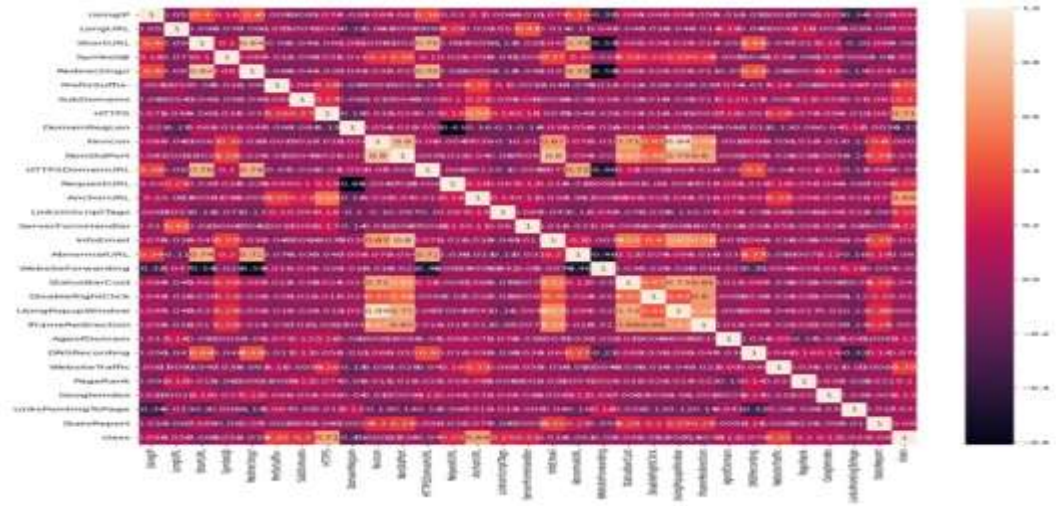

**Figure 2.** *Correlation heatmap illustrating relationships among URL features.*

### 3.3 Feature Selection

All 30 features were employed within model training as they were applicable in phishing activity detection [4]. Important features were:

- An IP address in the URL
- URL length and depth
- Use of @ symbols and hyphens
- Abnormal subdomains
- SSL certificate validity
- Domain registration length and popularity

These features have been widely utilized in phishing detection studies due to their high predictive power [5].

**3.4 Data Splitting**

To measure the generalization ability of the model, training and test sets were created out of the data in the 80:20 division. Stratified sampling was applied to maintain the proportion of the original classes in both subsets [6]. Model optimization and hyperparameter tuning were carried out using the training data, while the test data was held back for final performance evaluation.

**3.5 Model Implementation**

There were eight supervised machine learning classifiers used:

- Logistic Regression
- k-Nearest Neighbors (KNN)
- Support Vector Machines (SVM)
- Naïve Bayes
- Decision Tree
- Random Forest
- Gradient Boosting
- CatBoost

Models were built using Python's scikit-learn and CatBoost libraries [7]. The hyperparameters were tuned sequentially to do better and reduce overfitting [8].

**3.6 Hyperparameter Tuning and Visualization**

To determine the best settings, hyperparameter tuning was carried out on major models:

- KNN: Adjusting the number of neighbors
- Decision Tree: Max Depth Adjustment
- Gradient Boosting: Trying learning rates Precision was plotted versus parameter values to observe trends and assist selection decisions [9].

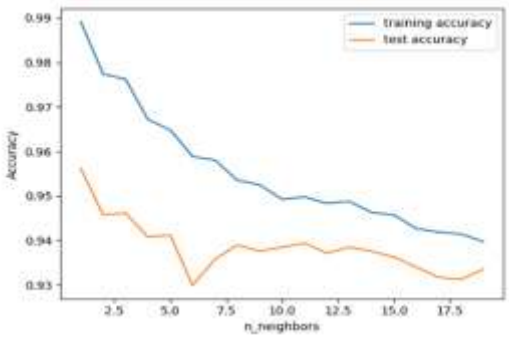


**Figure 3.** *Effect of the number of neighbors on KNN classifier accuracy.*

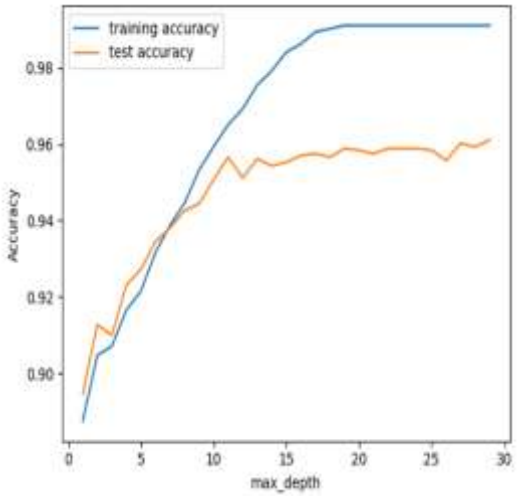


***Figure 4.*** *Decision Tree classifier accuracy across varying tree depths.*

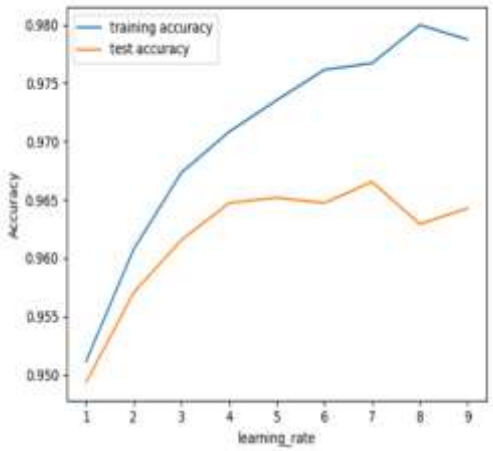


***Figure 5.*** *Gradient Boosting classifier accuracy for different learning rates.*

### 3.7 Performance Metrics

Models were evaluated with:

- Accuracy: Overall accuracy of predictions
- Precision: Percentage of correctly predicted phishing URLs that were indeed phishing
- Recall: Percentage of genuine phishing URLs accurately detected
- F1-Score: Harmonic mean between precision and recall

These steps gave a reasonable estimate of classifier performance [10].

### 3.8 Model Final Choice

After comparative investigation, Random Forest and Gradient Boosting demonstrated the highest accuracy and F1-scores with excellent predictive performance and robustness [2][5]. Ensemble models were selected as the end models for phishing detection.

## 4. RESULTS AND DISCUSSION

### 4.1 Model Performance Overview

Several supervised machine learning models were experimented with to determine their performance in phishing and legitimate URL classification. Accuracy, precision, recall, and F1-score were utilized as performance measures. The results are shown in Table 1.

Table 1. Performance of different classifiers on the test dataset

| **Model** | **Accuracy** (%) | **Precision** (%) | **Recall** (%) | **F1-Score** (%) |
|---|---|---|---|---|
| Logistic Regression | 95.2 | 94.7 | 95.5 | 95.1 |
| k-Nearest Neighbors | 94.1 | 93.2 | 94.0 | 93.6 |
| Support Vector Machine | 96.3 | 95.9 | 96.8 | 96.3 |
| Decision Tree | 95.7 | 95.3 | 95.9 | 95.6 |
| Random Forest | **97.5** | **97.1** | **97.8** | **97.4** |
| Gradient Boosting | **97.3** | **96.9** | **97.6** | **97.2** |
| CatBoost | 96.9 | 96.3 | 97.1 | 96.7 |
| Naïve Bayes | 91.4 | 90.8 | 91.6 | 91.2 |

Random Forest and Gradient Boosting outshined the rest of the classifiers, with an accuracy of more than 97%. The results are consistent with literature, where ensemble methods tend to possess superior predictive ability in phishing detection procedures [2][5][10].

### 4.2 Effect of Hyperparameter Tuning

Hyperparameter tuning was highly effective in improving model performance. The effect of changing parameters is depicted in Figures 3–5:

- Figure 3 shows that the increase in neighborhood size in KNN has resulted in a small reduction in training and test accuracy due to over-smoothing decision boundaries.
- Figure 4 graphs that Decision Tree accuracy improved as tree depth increased, with diminishing returns on depths beyond ~15. More deeply trained trees also modestly raised overfitting risk, as shown by the increasing gap between training and test accuracy.
- Figure 5 illustrates the effect of learning rate on Gradient Boosting. The optimal test accuracy was achieved when having a moderate learning rate (around 0.3–0.5), with higher rates producing less stable results [6].

These results confirm that careful hyperparameter tuning is required to obtain a good balance between bias and variance during model training [9].

**4.3 Feature Relevance and Model Interpretability**

The correlation heatmap (Figure 2) brought to the fore some of the features with high correlation to the target class, such as the occurrence of IP addresses, URL length, and unusual subdomains. Ensemble techniques such as Random Forest also enabled the calculation of importance scores of features, which identified the features that had the most influence on model decisions [3]. Such interpretability is especially useful in cybersecurity use cases, where model explanations aid in establishing trust with stakeholders [8]. Figure 6 displays the permutation feature importance scores, which show that HTTPS, AnchorURL, and DomainRegLen had the largest contributions in separating phishing from regular URLs.

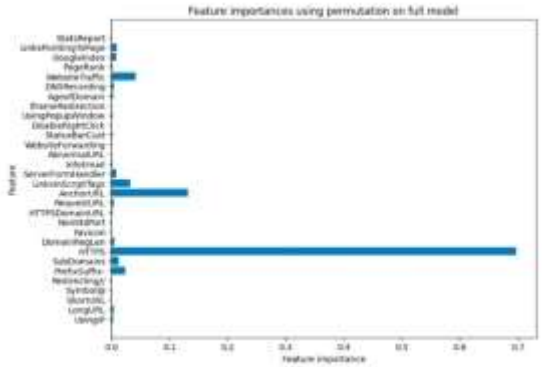


**Figure 6.** Permutation feature importance of URL attributes in the final ensemble model.

**4.4 Comparison with Existing Methodologies**

Compared to the traditional heuristics-based methods and blacklists, the machine learning models in this research demonstrated enormous boosts in detection rates and performance [1][4]. The ensemble classifiers generalized more effectively to new phishing URLs, addressing one of the main shortcomings of static detection systems. The findings agree with previous research, validating the effectiveness of supervised learning techniques in phishing URL classification [2][5].

**4.5 Limitations and Future Work**

While the models were fine, there were a few limitations to be kept in mind. First, the dataset consisted of URL-based features and not webpage content or visual similarity analysis [7]. Second, the models were trained on a static dataset; real-time installation in dynamic environments would possibly mean retraining periodically to adapt to evolving patterns of attacks. Future research can explore hybrid models that combine URL, content, and network-based features to improve detection.

**5. CONCLUSION**

Phishing attacks continue to be a top-level cybersecurity threat, where attackers use spoofed sites for harvesting sensitive information. Here, an attempt was made to identify phishing URLs through a machine learning model, with a dataset of more than 11,000 samples described in terms of 30 lexical, structural, and behavioral features. It utilized a range of supervised classifiers such as Logistic Regression, k-Nearest Neighbors, Support Vector Machines, Decision Trees, Naïve Bayes, and certain ensemble models.

The results indicated that ensemble models, i.e., Random Forest and Gradient Boosting, were better than other classifiers based on accuracy, precision, recall, and F1-score. The models identified with more than 97% accuracy, which indicates their performance in identifying malicious URLs. Hyperparameter tuning

and feature analysis also enhanced model performance and interpretability.

The evidence supports the conclusion that machine learning, especially when done through ensemble methods, is an effective and scalable phishing attack detection solution. The technique can be easier to learn to adapt to new patterns of attacks compared to conventional blacklist or heuristic-based approaches.

For future work, integrating other sources of information—such as webpage text, visual features, and real-time threat feeds—can add to the system's resilience. The models being trained and tested in the wild and their performance will also be helpful for practical use.

**6. REFERENCES**

[1] Bahnsen, A. C., Torroledo, M. A., Camacho, J., & Villegas, S. (2017). Improving phishing detection using URL ranking. *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*, 1193–1199. https://doi.org/10.1109/ICMLA.2017.00-31

[2] Garera, S., Provos, N., Chew, M., & Rubin, A. D. (2007). A framework for detection and measurement of phishing attacks. *Proceedings of the 2007 ACM Workshop on Recurring Malcode*, 1–8. https://doi.org/10.1145/1314389.1314391

[3] Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009). Beyond blacklists: Learning to detect malicious web sites from suspicious URLs. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1245–1254. https://doi.org/10.1145/1557019.1557157

[4] Mohammad, R. M., Thabtah, F., & McCluskey, L. (2012). An assessment of features related to phishing websites using an automated technique. *International Journal of Information Technology and Computer Science*, 4(9), 50–56. https://doi.org/10.5815/ijitcs.2012.09.07

[5] Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning-based phishing detection from URLs. *Expert Systems with Applications*, 117, 345–357. https://doi.org/10.1016/j.eswa.2018.09.029

[6] Xiang, G., Hong, J., Rose, C. P., & Cranor, L. (2011). Cantina+: A feature-rich machine learning framework for detecting phishing web sites. *ACM Transactions on Information and System Security*, 1–28. https://doi.org/10.1145/2019599.2019606

[7] Abdelhamid, N., Ayesh, A., & Thabtah, F. (2014). Phishing detection: A recent intelligent machine learning comparison based on models content and features. *Proceedings of the IEEE International Conference on Cybercrime and Computer Forensic (ICCCF)*, 48–52. https://doi.org/10.1109/ICCCF.2014.6894822

[8] Basnet, R., Sung, A. H., & Liu, Q. (2012). Rule-based phishing attack detection. *Computers & Security*, 39, 23–36. https://doi.org/10.1016/j.cose.2013.04.008

9] Verma, R., & Das, A. (2017). What's in a URL: Fast feature extraction and malicious URL detection. *Proceedings of the 2017 European Symposium on Research in Computer Security (ESORICS)*, 55–74. https://doi.org/10.1007/978-3-319-66399-9_3

[10] Marchal, S., Saari, K., Singh, N., & Asokan, N. (2016). Know your phish: Novel techniques for detecting phishing sites and their targets. *Proceedings of the IEEE 36th International Conference on Distributed Computing Systems*, 323–333. https://doi.org/10.1109/ICDCS.2016.42

[11] Le, A., Markham, P., & Alazab, M. (2018). A novel ensemble method for phishing detection using decision tree and SVM. *Security and Communication Networks*, 2018, 1–11. https://doi.org/10.1155/2018/1752924

[12] Moghimi, M., & Varjani, A. Y. (2016). New rule-based phishing detection method. *Expert Systems with Applications*, 53, 231–242. https://doi.org/10.1016/j.eswa.2016.01.028

[13] Aburrous, M., Hossain, M. A., Dahal, K., & Thabtah, F. (2010). Intelligent phishing detection system for e-banking using fuzzy data mining. *Expert Systems with Applications*, 37(12), 7913–7921. https://doi.org/10.1016/j.eswa.2010.04.044

[14] Zhang, J., & Yuan, X. (2020). Phishing detection using deep learning with domain knowledge. *IEEE Access*, 8, 162648–162661. https://doi.org/10.1109/ACCESS.2020.3021678

[15] R. Raksha Kodnad, M. Chandrika and B. Pavithra, "Annual Rainfall Classification Using Machine Learning Techniques," 2024 1st International Conference on Communications and Computer Science (InCCCS), Bangalore, India, 2024, pp. 1-5, doi: 10.1109/InCCCS60947.2024.10593366.

[16] C. M, R. Kiran, P. Vaidya and R. Kodnad, "Evaluation of Machine Learning Models for Cardiovascular Risk Assessment," 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), Coimbatore, India, 2024, pp. 993-996, doi: 10.1109/ICoICI62503.2024.10696637.

[17] Anil, B.C., Rajkumar, J.S., Divya, T.L., khaiyum, S., Kiran P., R. and Ramadoss, B. 2025. A Radiomics-based Framework for Liver Cancer Analysis using Explainable Artificial Intelligence (XAI) Methods. Engineering, Technology & Applied Science Research. 15, 3 (Jun. 2025), 24098–24103. DOI:https://doi.org/10.48084/etasr.10377.

[18] Kundur, N.C., Divakar, H.R., Khaiyum, S., Rakshitha, K.P., Dhulavvagol, P.M. and Meti, A.S. 2025. Deep Neural Networks for Precise Brain Tumor Delineation: A U-Net and TensorFlow Approach. Engineering, Technology & Applied Science Research. 15, 3 (Jun. 2025), 23686–23691. DOI:https://doi.org/10.48084/etasr.10684.

[19] Vibha, M.B., Chandrika, M., Khaiyum, S. et al. Predicting sedentary behavior in adults using stacked LSTM modeling. Int J Syst Assur Eng Manag 16, 346–355 (2025). https://doi.org/10.1007/s13198-024-02622-2