AVE Trends in Intelligent Computing Systems

Ensemble-Based Phishing Website Detection Using Extra Trees Classifier

M. Arjun Raj*, M. A. Thinesh, and S. S. Mukhil Varmann
Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
am0306@srmist.edu.in, tm9045@srmist.edu.in, sm7225@srmist.edu.in

Avinash Reddy Pothu
Department of Research and Development, Ginger Labs, Texas, United States of America.
Reddy0656@outlook.com

P. Paramasivan
Department of Research and Development, Dhaanish Ahmed College of Engineering, Chennai, Tamil Nadu, India.
paramasivanchem@gmail.com

*Corresponding author

Abstract: An attacker phishes victims to get their usernames, passwords, credit card numbers, or other personal information. Internet users are at risk from phishing attacks that steal personal data. Protecting against such dangers requires effective detection. Machine learning uses data-driven algorithms to detect phishing attempts and find patterns and abnormalities. Using the Extra Trees Classifier, the study investigates ensemble-based phishing website detection. A labelled dataset with phishing-related features trains and evaluates the proposed model. This research utilizes Kaggle's dataset, which has 89 URL-, content-, network-, and statistical attributes. These traits help the model distinguish phishing websites from authentic ones. Explore and visualize these features to understand data distribution and feature relationships. The dataset is separated into training and testing datasets after visualization and used for model training and testing. The model is evaluated using ExtraTrees Classifier with 96.68% accuracy, 97.65% precision, 95.58% recall, and 96.6% F1 score. The project introduces a strong online user protection method based on machine learning for phishing detection. The project was developed using Google Collab.

Keywords: Machine Learning; URL-Based Features; Extra Trees Classifier; Ensemble Learning; Phishing Detection; Social Media; Training and Evaluating.

Cite as: M. A. Raj, M. A. Thinesh, S. S. M. Varmann, A. R. Pothu, and P. Paramasivan, "Ensemble-Based Phishing Website Detection Using Extra Trees Classifier," *AVE Trends In Intelligent Computing Systems*, vol. 1, no. 3, pp. 142 –156, 2024.

Journal Homepage: https://avepubs.com/user/journals/details/ATICS

Received on: 17/02/2024, **Revised on:** 07/04/2024, **Accepted on:** 03/06/2024, **Published on:** 01/09/2024

1. Introduction

The Internet has impacted society in many different ways, such as in business, education, and social interactions. The Internet has both positive and negative sides to it. The Internet has changed how people socialize. Social media is now used to connect to different kinds of people, but it also contains most of the personal information that could be used for bad purposes. The most important aspect of society that the Internet has transformed is the business industry. Tasks once, which were time-consuming, are now easily done [15]. E-commerce has grown exponentially, allowing many businesses to reach the global market [16]. The increase in the use of social media has enabled many people to start small-scale businesses and easily connect with the people they are looking for [17]. It also enabled the use of digital payment, which most people use as they don't need to carry any physical money, and they don't need to be scared of losing their money or getting their money stolen. However, this also created concerns about data privacy and many malicious websites and applications [18]. The increase in the usage of the Internet

also leads to an increase in cybercriminals. Malicious websites are websites that tend to steal the users' private data and use it maliciously [19]. Phishing is one of the malicious methods used by cybercriminals to steal the user's private data and use it to exploit money from them or sell those data. Most phishing is done by using a website or a mail that looks trusted and deceives the user to enter their details [20].

Phishing is one of the most common forms of cybercrime. Phishing is a type of cybercrime where criminals attempt to deceive the user into providing their details such as usernames, passwords, credit card numbers and other details. In 2021, globally, 323,972 internet users fell victim to phishing attacks [21]. In 2022, the most used URL included in phishing email links to websites with the '.com' domain at 54% and the next most commonly used domain is '.net' at less than 8.9%. The domains '.com' and '.net' are the most commonly used domains for any trusted websites, so people don't think of the websites with '.com' and '.net' domains as malicious websites and trust them [22]. Still, cybercriminals use this to their advantage and deceive people to get their data. Phishing is mostly done by using spam mail [23]. These spam emails are made to look trustworthy and make the user click the link, which leads to a website that looks authentic and allows the user to enter their data into the website[24].

There are also other types of phishing, such as whaling, smishing, and vishing. In whaling, the attackers target high-profile individuals in an organization, such as executives or CEO. Smishing involves sending fraudulent messages, and vishing uses phone calls [25]. All this phishing is done to trick people into getting their personal information. Phishing emails may also contain attachments that, when downloaded, will install malware on the device, and the malware will be used to steal the user's data, such as keystrokes [26]. Cybercriminals deceive people to get their details and also deceive them into getting their OTP, which is necessary to transfer money using Internet banking [27]. These phishing emails are mostly blocked as spam mail, but some emails look so authentic that the AI cannot detect them, so in those cases, it is the responsibility of humans to be aware of them [28]. The best way to get tricked by phishing is to become aware of it and learn how to recognize and respond to phishing attempts. By understanding the nature of phishing and implementing defence mechanisms, individual people and organizations can better protect themselves against these phishing attacks [29].

Machine learning has become an important technology in the fight against phishing, particularly in the detection of URLs that are phishing [30]. The traditional techniques have shown insufficient ability to keep up with the fast-changing character of phishing attacks. Machine learning techniques provide a strong and flexible method that is capable of analyzing various features and identifying patterns that signal phishing threats [31]. The machine Learning model's adaptability allows them to learn from new data and change to fit developing phishing techniques [32]. Machine Learning models can achieve great accuracy in identifying phishing URLs. Attackers continually develop new evasion techniques, thus making regular updates and improvements to the model necessary [33]. Ensemble learning is a very powerful machine learning technique which combines the predictions of multiple models to improve the accuracy of the prediction [34].

Ensemble learning is used because of the idea that by aggregating the predictions of multiple models, ensemble learning can perform better than individual models and give us better and more accurate predictions. The way in which the predictions of the models are combined is crucial [35]. This can be done in various ways, such as 'voting' for classification, 'averaging' for regression, or using a meta-model to learn how to combine the predictions (stacking) effectively. Using the strengths of every model will help to minimize their weaknesses [36]. Ensemble methods are computationally complex, but their ability to outperform individual models makes them a better choice for complex problems. As the detection of phishing is a complex problem, the ensemble method will perform better than an individual model. One of the ensemble learnings is the ExtraTrees classifier, which is an extension of the Random Forest classifier and has a collection of decision trees. Each tree is trained on different subsets of the data. It introduces additional randomness in the tree-building process compared to the use of random forests [37]. This additional randomness is used to reduce the variance in the model, making it less likely to overfit and thus enhancing performance [38]. The ExtraTrees classifier is effective in scenarios where the features have complex interactions. ExtraTrees classifier is used for phishing detection due to its high performance and ability to handle large datasets [39].

The continuous evolution of phishing methods calls for constant research and development in this field to guarantee that the detection systems can detect the newly invented methods and reduce the number of users getting attacked by phishing [40]. Users need to have some knowledge about phishing attacks and spam mail so that they can avoid these malicious attacks [41]. They should only use links from trustworthy sources and always check if the website is secure or not before giving their details. The models can be used to detect phishing attacks, but due to newly updated malicious techniques, the model may not detect them, and the model must be updated frequently with new data to overcome this problem.

2. Objective

- The goal of developing this model is to detect and classify phishing attacks using URLs. This includes collecting and pre-processing data, selecting necessary features, and training and evaluating the model's performance.
- The project aims to build a robust model that can accurately detect phishing attempts, thereby enhancing network security and reducing the risk of phishing attacks.

- Implementing this model in systems will enable real-time detection of phishing attacks, thereby protecting users from malicious URLs and potential data breaches.

3. Literature survey

Karim et al. [1] focus on developing a hybrid machine-learning model for phishing detection using URLs, addressing the increasing threat of phishing attacks in the digital landscape. The study highlights the evolution of phishing since 1996, emphasizing its significance as a major cybercrime. The proposed model achieves an impressive accuracy of 98.12%, surpassing previous methods. It utilizes techniques like Grid search with cross-fold validation and canopy feature selection to enhance performance. The findings underscore the importance of effective phishing detection systems in safeguarding internet users. Overall, the research contributes valuable insights to the field of cybersecurity.

Ali and Ahmed [2] propose a hybrid model for phishing website detection. It combines deep neural networks (DNN) with genetic algorithms (GA) to select and optimize website features. The model improves accuracy by using GA to identify the most relevant features and their optimal weights, which are then used to train DNNs. Experimental results show significant improvements in classification accuracy, sensitivity, and specificity compared to other methods.

Jibat et al. [3] review the methods used to detect phishing websites. The research covers data mining and machine learning techniques published between 2018 and 2021. It examines how algorithms like Random Forest, Naive Bayes, and Support Vector Machines are applied, with many achieving over 90% accuracy. While some models perform well, none reach 100% accuracy in detecting phishing websites. The study highlights the importance of feature selection and the ongoing need for improvements in phishing detection models.

Zhu et al. [4] introduce a phishing detection model that combines a revised multi-objective evolution optimization algorithm (MOE) with a random forest classifier (RF). This hybrid model optimizes feature selection and classification simultaneously, improving both detection accuracy and computational efficiency. By addressing multiple objectives, such as maximizing detection rates and minimizing false positives, the model offers a more effective solution for identifying phishing websites in complex network environments.

Kara et al. [5] explore the use of machine learning techniques to detect phishing websites by analyzing URL and domain name features. The study identifies key characteristics that differentiate phishing from legitimate websites, improving detection accuracy. By leveraging various machine learning algorithms, the authors propose a model that effectively identifies phishing attempts based on URL structure and domain patterns, contributing to stronger cybersecurity defences.

Kalabarige et al. [6] present a robust phishing detection framework that integrates hybrid feature selection and a multi-layer stacked ensemble learning model. By using boosting techniques, the model enhances classification performance and accuracy, effectively distinguishing between phishing and legitimate websites. The hybrid feature selection optimizes the identification of relevant features, while the multi-layer stacked ensemble combines various machine learning models to improve overall detection capabilities, making it a powerful tool in combating phishing threats.

Tang and Mahmoud [7] present a phishing detection model that leverages deep learning techniques to identify malicious websites. The framework utilizes advanced neural networks to automatically extract and learn features from website URLs and associated content, improving detection accuracy. By applying deep learning, the model outperforms traditional machine learning methods in recognizing complex phishing patterns, providing a scalable and effective solution to combat phishing attacks across various digital platforms and environments.

Dhinakaran et al. [8] explore the use of ensemble learning techniques to improve cyber intrusion detection systems. The authors compare bagging and stacking classifiers, evaluating their performance in identifying security threats. The study demonstrates that both methods enhance detection accuracy by combining multiple models, but stacking provides more robust performance. This research highlights the potential of ensemble approaches in strengthening cybersecurity measures by detecting complex intrusion patterns.

Yang et al. [9] propose a phishing detection model that utilizes deep learning to analyze multidimensional features of websites. By incorporating diverse characteristics such as URL structures, website content, and domain-related information, the model improves accuracy in distinguishing phishing sites from legitimate ones. The deep learning framework automates feature extraction and learns complex patterns, offering an effective and scalable solution for identifying phishing websites, ultimately enhancing online security and user protection.

Zieni et al. [10] provide a comprehensive review of phishing detection techniques. It categorizes detection methods into three approaches: list-based, similarity-based, and machine learning-based. The paper covers the evolution of phishing tactics and the importance of detecting phishing websites through URL, content, and visual analysis. Additionally, it discusses the strengths and limitations of various methods and highlights research gaps that need to be addressed to improve phishing detection models.

Al-Ahmadi et al. [11] introduce a novel phishing detection model leveraging Generative Adversarial Networks (GANs). The model uses the GAN framework to generate synthetic phishing data, which improves the training of phishing detection systems by enhancing feature diversity and coverage. By utilizing both the generative and discriminative capabilities of GANs, the PDGAN model improves detection accuracy and robustness against evolving phishing techniques, offering an advanced approach to strengthen cybersecurity defences.

Mohanty et al. [12] present a hybrid feature selection technique aimed at improving the prediction of suspicious URLs within the Internet of Things (IoT) environments. The proposed model combines different feature selection methods to identify the most relevant features, enhancing the detection accuracy while reducing computational costs. By focusing on IoT-specific challenges, the model improves the identification of malicious URLs, contributing to enhanced security in IoT networks against cyber threats.

Alsariera et al. [13] introduce a model for detecting phishing websites using AI meta-learners combined with the Extra-Trees algorithm. By employing ensemble learning, the authors aim to improve the accuracy and reliability of phishing detection. The study analyzes various website features to identify patterns typically associated with phishing. The Extra-Trees algorithm, known for its efficiency in handling high-dimensional data, is integrated with meta-learning techniques to boost performance. It offers a scalable and effective solution for cybersecurity challenges in detecting phishing threats.

Wei and Sekiya [14] explore the effectiveness of ensemble machine-learning methods for detecting phishing websites. They evaluate the sufficiency of various ensemble techniques, such as bagging, boosting, and stacking, to enhance detection accuracy. By analyzing website features and comparing ensemble methods against individual classifiers, the study demonstrates how these techniques improve phishing detection performance. The results highlight the potential of ensemble models as reliable solutions for identifying phishing threats in real-time cybersecurity applications.

4. Proposed methodology

The proposed methodology covers various stages, including data pre-processing, feature engineering, model selection, training, and evaluation. Initially, we gathered a dataset from a trustworthy source, encompassing numerous attributes that are indicative of phishing websites. These attributes include the length of the URL, the number of special characters, the presence of IP addresses within the URL, and other relevant domain-related features. The selection of these features was done with great care, ensuring their relevance and potential impact on the model's ability to differentiate between phishing and legitimate websites.

During the data pre-processing phase, the dataset underwent multiple transformations to make it suitable for model training. This involved addressing any missing values, normalizing numerical features, encoding categorical variables, and possibly creating new features that could capture more complex patterns within the data. Subsequently, the dataset was divided into training and testing subsets through a stratified approach, preserving the distribution of the target variable across both subsets. This step was crucial in preventing data leakage and ensuring that the model's performance could be accurately assessed on unseen data, thereby providing a more reliable evaluation of its generalization capabilities. In the feature selection stage, we identified the most significant attributes that contribute to the model's predictive ability.

In this approach, the process of selecting features played a significant role, incorporating techniques like recursive feature elimination (RFE), assessing feature importance through ensemble methods, and conducting correlation analysis to eliminate irrelevant or redundant variables [42]. This ensured that only the most influential features were used for model development. These refined features were then employed to train the proposed Extra Trees Classifier, a method that combines predictions from multiple decision trees to enhance both precision and robustness [43]. The Extra Trees Classifier was chosen for its ability to effectively manage high-dimensional datasets and reduce variance by utilizing bootstrapping and random feature selection. The training phase involved applying the Extra Trees Classifier to the training data, allowing the model to identify patterns between the input features and the target variable, which classifies websites as either legitimate or phishing. The phase includes prediction by voting based on the tree weights [44].

This methodology ultimately resulted in a model capable of delivering accurate and reliable phishing detection. After the model was trained, it was evaluated on the test set using various metrics, including accuracy, precision, recall, F1-score, and the area under the ROC curve (AUC-ROC), to assess its performance. Of particular interest was the accuracy metric, which indicated the proportion of correct predictions made by the model out of all predictions [45]. In this project, the model achieved an accuracy rate of 96.68%, demonstrating its effectiveness in detecting phishing websites [46]. Additionally, the model's probabilistic outputs, obtained through the predict_proba function, were analyzed to understand the confidence levels of its predictions and to explore potential adjustments to thresholds that could improve the balance between precision and recall. To further refine the model's performance, various optimization strategies were considered [47]. These included feature engineering techniques such as polynomial feature expansion, the introduction of interaction terms, and the integration of domain-specific knowledge that could lead to the creation of new features. Furthermore, ensemble methods like stacking or blending, where multiple models are combined to enhance overall performance, were explored [48]. The final model was then

deployed in a simulated or real-world environment to evaluate its ability to detect phishing websites in a dynamic and potentially adversarial setting [49].

The entire methodology was meticulously documented, capturing the rationale behind each step, the challenges encountered, and the decisions made to overcome those challenges. This documentation serves not only as a guide for future improvements but also provides the transparency and reproducibility necessary for maintaining the scientific rigour of the project [50]. Thus, the proposed methodology represents a comprehensive, iterative process designed to develop a highly accurate and reliable phishing detection model that can be effectively deployed in real-world applications, ultimately contributing to cybersecurity.

4.1. Architecture diagram

Figure 1 shows the utilization of an advanced model known as the Extra Trees Classifier, which enhances the capabilities of traditional Decision Trees [51]. This approach was chosen to overcome the shortcomings of the initial Decision Tree model, which, while useful in some contexts, tends to overfit when dealing with more complex data [52]. Overfitting often arises when a single tree analyzes the entire dataset and evaluates all available features at each decision point [53]. This may lead to learning patterns that reflect noise rather than meaningful trends [54].

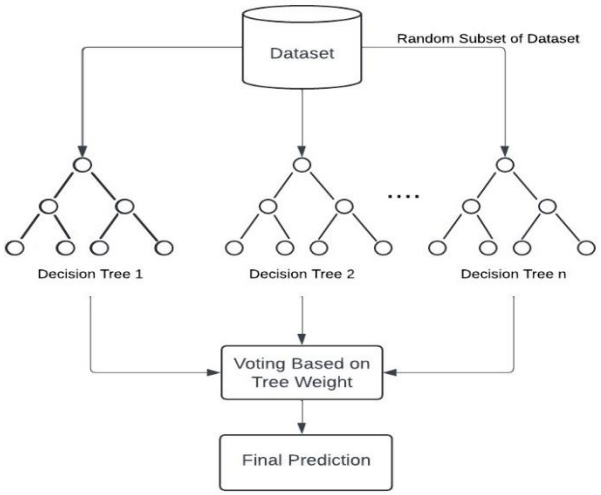

Figure 1: Model architecture of ExtraTrees Classifier

To address these issues, the Extra Trees Classifier uses an approach called extremely randomized trees, where multiple decision trees are trained using randomly chosen subsets of the dataset [55]. At each decision node, random splits are made rather than selecting the feature with the highest discriminative power, as is typical in traditional decision trees that rely on measures like Gini impurity or information gain [56]. This randomness adds a layer of diversity among the trees, helping to reduce correlations and making the model more adaptable to fluctuations in the dataset. After training, the model aggregates predictions from all the individual trees using a voting process based on individual tree weights for classification problems. This illustrates the superiority of ensemble learning techniques such as Extra Trees in managing complex datasets, particularly those associated with phishing detection, which often involve noisy and high-dimensional data [57].

The Extra Trees Classifier offers several advantages over the basic Decision Tree model, including enhanced resistance to overfitting, faster processing due to randomized splits, and the ability to efficiently manage a large number of features. These attributes make the Extra Trees model particularly suited for applications in cybersecurity, such as phishing detection, where distinguishing between legitimate and malicious sites requires both accuracy and the ability to avoid false positives.

4.2. Algorithm

- Load the Phishing Dataset with features related to phishing and legitimate websites.
- Data Transformation: Standardize and normalize numerical features and encode categorical features.
- Feature Selection: Select features using RFE
- Visualize Data: Use scatter plots, correlation matrices, and histograms to explore feature relationships and detect outliers.
- Split Data into training and testing sets, ensuring a balanced distribution of the target variable.
- Build and Evaluate Model:
    - Choose evaluation metrics like accuracy, precision, recall, and F1-score.

- Train the Extra Trees Classifier on the training data.
- Evaluate the model on testing data using chosen metrics.

- Generate Predictions: Use the Extra Trees Classifier to predict phishing or legitimate URLs for new data.
- Evaluate Model Performance: Assess accuracy, precision, recall, F1-score, and AUC-ROC on unseen data.
- End

4.3. Formulas

$$\text{Accuracy} = \frac{\text{Total Number of Predictions}}{\text{Number of Correct Predictions}} \times 100\%$$

• True Positives are the cases correctly predicted as positive.
• False Positives are the cases incorrectly predicted as positive

Accuracy represents the proportion of correctly classified instances out of the total instances. It is expressed as a percentage, indicating the model's overall correctness in its predictions.

$$\text{Precision} = \frac{\text{True Positives + False Positives}}{\text{True Positives}} \times 100\%$$

Precision measures the proportion of true positive predictions out of all positive predictions made by the model. It is expressed as a percentage, indicating the model's ability to avoid falsely labelling negative instances as positive.

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision + recall}}$$

F1-Score is the harmonic mean of precision and recall, providing a balance between the two metrics. It is a single value that considers both precision and recall, making it a useful metric for evaluating the overall performance of a classification model.

4.4. Existing model

The decision tree algorithm operates by progressively splitting the data based on various feature values, resulting in a tree structure. Each internal node represents a decision point tied to a particular feature, while the leaves of the tree correspond to the final predicted class labels. This hierarchical model structure allows for efficient mapping of input features to their respective outcomes, making it particularly suitable for simpler classification tasks. However, despite its benefits, the Decision Tree model has notable drawbacks, the most significant being its inclination to overfit the training data. Overfitting happens when the model becomes overly specific to the training data's unique characteristics, including noise, thus reducing its ability to generalize to new, unseen data. This issue is particularly acute when dealing with more complex, high-dimensional, or noisy datasets—conditions that are often present in phishing detection. Although Decision Trees can effectively identify decision boundaries, they can also pick up on irrelevant patterns, which leads to underwhelming performance when the model is tested on new datasets.

Furthermore, Decision Trees are deterministic, meaning that at each split, the algorithm selects the most informative feature based on metrics like Gini impurity or information gain. While this method may work well for smaller or well-structured datasets, it tends to be less effective when faced with larger datasets that contain numerous features, as not all selected features are optimal for generalizing. As a result, although the Decision Tree model served its purpose for initial experimentation, it struggled to deliver strong results when addressing the complex and ever-evolving characteristics of phishing datasets. The Decision Tree model's vulnerability to overfitting, particularly when working with phishing detection data, led us to explore more advanced methodologies. This prompted the transition to an ensemble-based approach, which we determined would be better equipped to handle the complexities of phishing detection.

4.5. Execution

The implementation was executed with a structured approach, adhering to a well-defined methodology to ensure both precision and robustness. The process began with meticulous pre-processing of the dataset, which involved several crucial steps, including data cleaning, transformation, and feature engineering. These steps were essential in preparing the data for the model by resolving any inconsistencies, standardizing numerical features, encoding categorical data, and crafting additional features that could potentially enhance the model's predictive power. Following this, the dataset was divided into training and testing sets to enable an impartial assessment of the model's performance.

The Extra Trees Classifier, an ensemble learning technique known for its effectiveness in handling complex, high-dimensional data, was selected for model training. This method was chosen because it excels in reducing variance through the collective decision-making of multiple trees. The classifier was trained on the carefully pre-processed data. Upon completing the training, the model's performance was tested on the test set, where it achieved a remarkable accuracy rate of 96.68%, confirming its proficiency in identifying phishing websites.

Throughout the execution process, several evaluation metrics were computed, including precision, recall, F1-score, and AUC-ROC. These metrics provided a comprehensive overview of the model's performance, highlighting both its strengths and areas where further improvement might be needed. The entire execution phase was thoroughly documented, capturing every decision and action to ensure the reproducibility of the project. This detailed execution not only validated the effectiveness of the chosen methodology but also demonstrated the model's practical application in real-world cybersecurity scenarios.

5. Implementation

5.1. Data and pre-processing

This dataset comprises 11,430 entries with 89 features, each representing various characteristics of URLs to aid in identifying phishing attempts. The features can be categorized into different types, including those based on length, such as length_url and length_hostname, which measure the length of the URL and hostname, respectively. Character count features like nb_dots, nb_hyphens, nb_at, and nb_qm count the occurrences of specific characters in the URL, often manipulated in phishing URLs to resemble legitimate websites.

The dataset also includes binary features indicating the presence or absence of specific elements within the URL, such as http_in_path, which checks if "http" is present in the path; https_token, indicating an "https" token in an unusual context; and punycode, checking if the URL uses punycode encoding. These elements are commonly used in phishing URLs to deceive users. Additionally, domain-related features such as domain_registration_length, domain_age, and web_traffic provide information about the domain's characteristics, with phishing domains often being recently registered and having low web traffic. Other features include port, tld_in_path, tld_in_subdomain, ratio_digits_url, ratio_digits_host, and more, offering further insight into the URL's structure and composition. The target variable, status, indicates whether a URL is classified as "legitimate" or "phishing".

During the pre-processing phase, this target variable was encoded into numerical values to facilitate model training, mapping "legitimate" to 0 and "phishing" to 1. This transformation makes the categorical target variable suitable for machine learning algorithms. Initial data exploration revealed that the dataset does not contain any missing values, ensuring completeness and eliminating the need for imputation or removal of entries due to missing data, thereby simplifying the pre-processing workflow.

To further prepare the data for model training, the feature set was standardized to ensure that each feature contributes equally to the model's performance by scaling the data to have a mean of zero and a standard deviation of one. This step is crucial for algorithms like the Extra Trees Classifier, which can be sensitive to the scale of input features. In summary, the pre-processing steps involved encoding the target variable, verifying the absence of missing values and standardizing the feature set, ensuring the dataset was in optimal condition for training the Extra Trees Classifier model, which subsequently achieved an accuracy of 96.68% in detecting phishing URLs.

5.2. Data visualization

Figure 2 offers a detailed visualization of the relationships between various features in the dataset and the target variable, status_encoded. Each point in this plot represents a URL, with colours distinguishing between "legitimate" (encoded as 0) and "phishing" (encoded as 1) URLs. Key features illustrated include length_url (the URL's length), length_hostname (the hostname's length), nb_dots (the number of dots in the URL), and nb_hyphens (the number of hyphens in the URL). Observations from this visualization reveal that diagonal plots indicate right-skewed distributions for features like length_url and length_hostname, suggesting that most URLs and hostnames are relatively short, with a few significantly longer ones. Similarly, features like nb_dots and nb_hyphens also show right-skewed distributions, indicating that most URLs contain fewer dots and hyphens.

Positive correlations are evident in the off-diagonal plots, such as between length_url and length_hostname, suggesting that longer URLs often have longer hostnames. The colour-coded points clearly illustrate the separation of classes, showing that URLs with higher numbers of dots or hyphens are more prevalent among phishing URLs. This indicates that these features are valuable for distinguishing between legitimate and phishing URLs.

Figure 2: Pair plot of the URL Dataset

Additionally, scatter plots highlight clusters of phishing URLs at higher values of nb_dots and nb_hyphens, as well as potential outliers, such as URLs with exceptionally high lengths, which may require further analysis. Overall, this pair plot offers significant insights into the dataset's structure, demonstrating how certain features can effectively separate legitimate URLs from phishing ones. This supports their use in the Extra Trees Classifier model and aids in understanding data distribution, identifying correlations, and spotting outliers, all of which are essential for developing a robust phishing detection model.

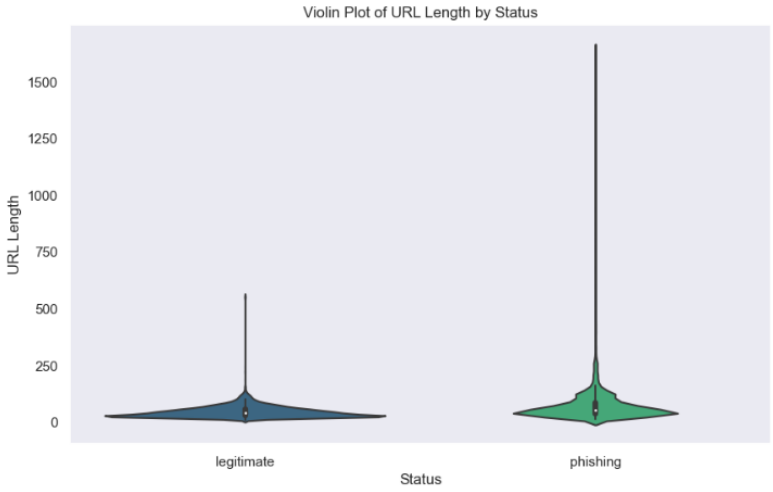

Figure 3: Violin Plot of URL by Status

Figure 3 illustrates the variations in URL lengths for both legitimate and phishing websites. The x-axis classifies the URLs into two groups: "legitimate" and "phishing," while the y-axis represents the URL length. The plot reveals that legitimate URLs tend to be shorter and more uniform in length. This is evident from the compact shape of the violin plot on the left. Legitimate URLs are mostly clustered around a shorter length, suggesting that legitimate websites typically use concise URLs. The median URL length for legitimate websites is indicated by a white dot within the violin plot, further supporting the observation that these URLs are generally shorter and exhibit less variation in length. Conversely, the distribution of phishing URLs, shown on the right side of the plot, spans a much wider range of lengths. The elongated shape of this violin plot signifies that phishing URLs can vary greatly, with some URLs being quite lengthy. This variation suggests that phishing websites often employ longer and more complex URLs, possibly to conceal their malicious nature or to better mimic legitimate URLs. The white dot, representing the median length of phishing URLs, is positioned higher than that of legitimate URLs, indicating that phishing URLs tend to be longer on average. In summary, this plot visually demonstrates the differences in URL lengths between

legitimate and phishing websites. The distinct variation in URL length distributions can be a helpful indicator for detecting phishing attempts, as longer and more variable URLs are more frequently associated with phishing websites.

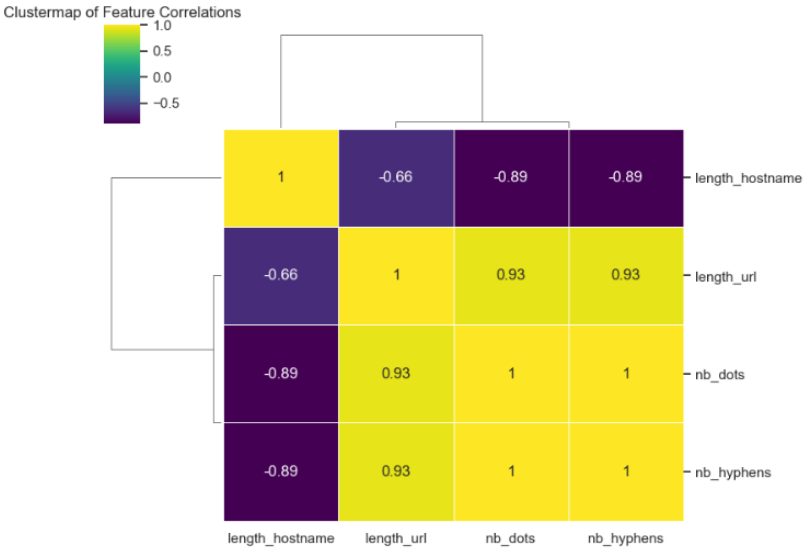

Figure 4: Cluster of Feature Correlation

Figure 4 provides an analysis of the relationships between various URL characteristics, such as the hostname length, URL length, number of dots, and number of hyphens. The intensity of the colours signifies the strength and direction of these relationships, with dark purple showing strong negative relationships and bright yellow indicating strong positive ones. Some key insights include a moderate negative relationship between hostname length and URL length (-0.66) and strong negative relationships between hostname length and the number of dots (-0.89) and hyphens (-0.89). This implies that longer hostnames are often associated with shorter URLs that have fewer dots and hyphens. On the other hand, the URL length has a strong positive relationships with both the number of dots (0.93) and hyphens (0.93), suggesting that longer URLs generally contain more dots and hyphens. Furthermore, there is a perfect positive relationship (1) between the number of dots and hyphens, indicating that URLs with more dots tend to also have more hyphens. This map clearly shows how these URL features are connected, offering valuable insights for evaluating and identifying potential phishing URLs.

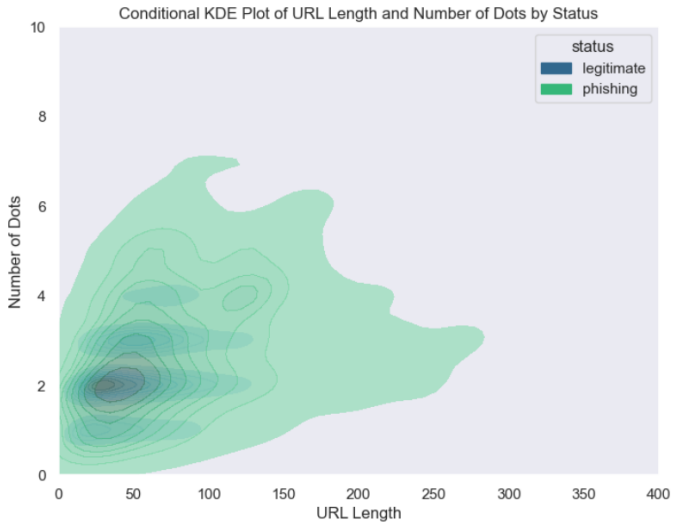

Figure 5: Conditional KDE plot of URL Length and Number of Dots by Status

Figure 5 provides a nuanced visualization of the relationship between URL length and the number of dots within a URL, categorized by their status as legitimate or phishing. This plot serves as an essential tool in understanding the subtle differences and overlaps between these two categories. The KDE contours demonstrate that while legitimate and phishing URLs share some common characteristics, they diverge significantly in certain aspects. Legitimate URLs predominantly cluster around shorter lengths with fewer dots, suggesting a more straightforward structure typical of genuine websites. In contrast, phishing URLs display a broader and more dispersed distribution, particularly extending into longer URL lengths with a higher number

of dots. This pattern suggests that phishing URLs often adopt more complex and elongated structures, potentially as a tactic to obfuscate their malicious intent and avoid detection by users and automated systems.

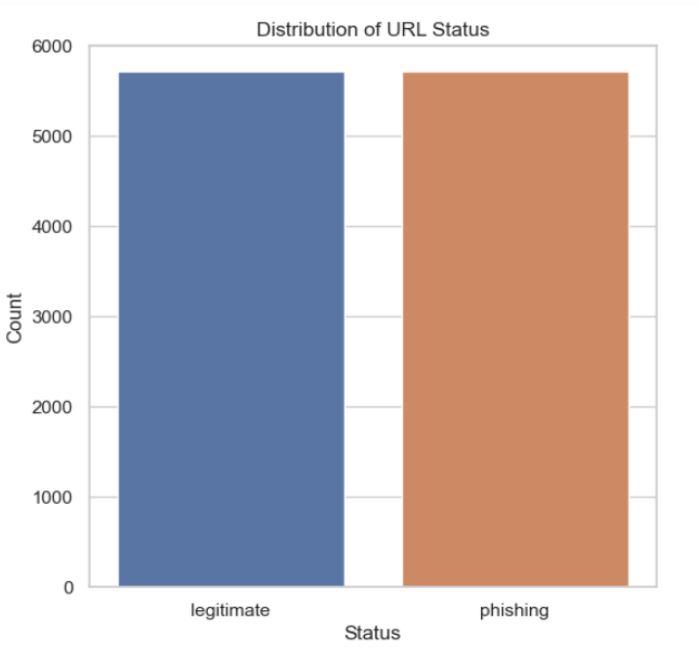

Figure 6: Distribution of URL status

Figure 6 displays the distribution of URLs categorized as either legitimate or phishing. The count for each category is approximately equal, with both legitimate and phishing URLs numbering around 5,500. This balanced distribution suggests that the dataset used for this analysis provides an even representation of both types of URLs. A dataset like this is particularly valuable for training machine learning models, as it minimizes the risk of bias towards one category. Consequently, the model is better equipped to accurately differentiate between legitimate and phishing URLs in practical applications. This balanced dataset serves as a strong foundation for conducting further analysis and building effective detection mechanisms.

5.3. Training

The training phase of this project was pivotal in developing a robust model capable of accurately detecting phishing websites. During this phase, the pre-processed dataset, which had been carefully curated to include the most relevant features, was used to train the Extra Trees Classifier. This ensemble learning algorithm was chosen for its ability to handle high-dimensional data and its resilience to overfitting, achieved by aggregating the predictions of multiple decision trees. The model was trained on the training set, with hyperparameters such as the number of trees, maximum depth, and minimum samples per split meticulously tuned to optimize performance. To ensure that the model generalized well to unseen data, cross-validation techniques were employed, allowing the model to be evaluated across different subsets of the data. This approach helped in fine-tuning the model and ensuring that it did not just memorize the training data but rather learned the underlying patterns that differentiate phishing from legitimate websites. The result was a well-calibrated model that demonstrated high accuracy and reliability when applied to the test data, reinforcing the effectiveness of the training process.

6. Results and Discussion

In this paper, we chose Python to develop our CatBoost Classifier model. The proposed model was run and evaluated on Windows 11 with AMD Ryzen 7 5800x, 32 GB RAM, RTX 3060 Ti using Jupyter using Python language. Jupyter Notebook is a powerful software used for training machine learning models, and using big datasets is better in Jupyter Notebook. The dataset is used to train the proposed ExtraTrees model.

The dataset is also used to train multiple models such as Decision Tree, AdaBoost, Bagging, and RNN and is compared with the ExtraTrees model. The model is validated using a tested dataset. The proposed model detects and classifies whether the website is legitimate or phishing. The model is evaluated using parameters including Accuracy, Average Precision, Average Recall, and Average Recall. The model is also evaluated using the ROC Curve. A Learning Curve is used to compare training accuracy and validation accuracy.

Table 1: Comparison of Result Metrics

Model	Accuracy	Precision	Recall	F1-Score
Decision Tree	92.73%	92.78%	92.51%	92.65%
AdaBoost	95.13%	95%	95.16%	95.08%
Bagging	95.5%	96%	94.8%	95.43%
RNN	95.42%	95.34%	95.40%	95.37%
ExtraTrees	96.68%	97.65%	95.58%	96.6%

Table 1 provides a comparative analysis of various machine learning models used in phishing detection, evaluated on four performance metrics: Accuracy, Precision, Recall, and F1-Score. The models tested include Decision Tree, AdaBoost, Bagging, Recurrent Neural Network (RNN), and ExtraTrees. The ExtraTrees model yielded the highest accuracy of 96.68%, with a Precision, Recall, and F1-Score of 97.65%,95.58% and 96.6%, respectively, demonstrating its robustness and efficiency in detecting phishing URLs. Among all the models, ExtraTrees consistently outperformed the others, showcasing its superior capability in this domain.

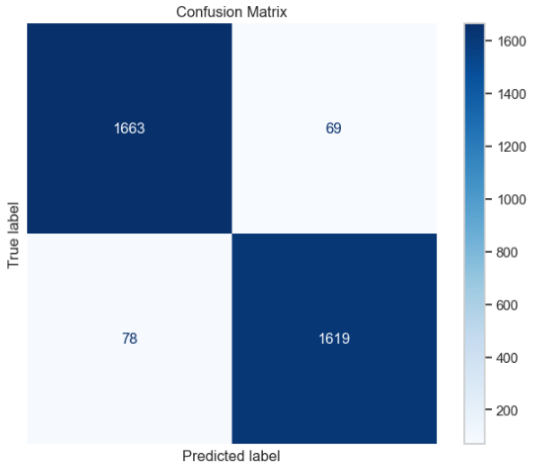

Figure 7: Confusion Matrix of ExtraTrees Model

Figure 7 is the confusion matrix of the proposed model. In statistical classification and machine learning, the confusion matrix is an invaluable instrument for assessing the performance of a classification model. In addition to allowing you to view the frequency with which your classifier is accurate, it also enables you to observe the different kinds of mistakes that it produces. For a binary classification problem, the confusion matrix is a 2x2 table with four possible outcomes, i.e., True Positive, True Negative, False Positive and False Negative. Confusion matrix can be used to calculate accuracy, precision, recall, specificity, F1 Score, False Positive Rate and False Negative Rate.

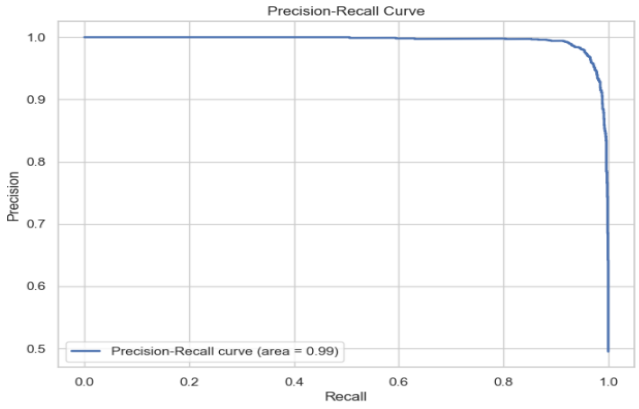

Figure 8: Precision-Recall Curve of Extra Trees Model

Figure 8 is the Precision-Recall Curve generated for the model. The precision-recall curve is a graphical representation of the model's precision and recall score. The precision-recall curve is a very useful method to evaluate the performance of a classification model. The precision-recall curve is created by plotting the precision of the model against the recall for different thresholds used by the model to make predictions. If the area under the curve (AUC) is high, it means that the model is highly accurate. If the AUC is low, it means the model is not doing very well. The AUC is 0.99 in this graph, which is very close to 1, suggesting excellent model performance.

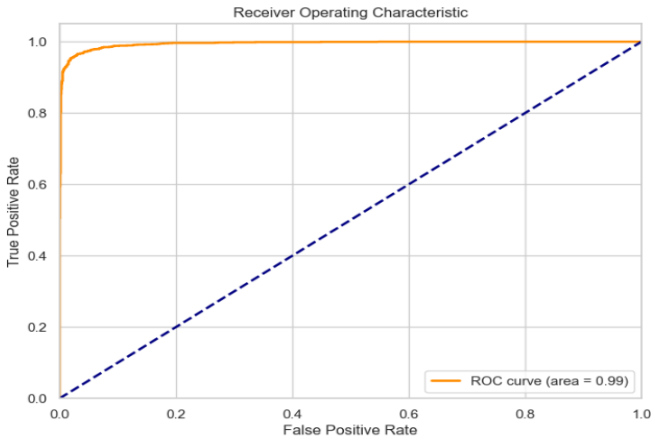

Figure 9: ROC Curve of ExtraTrees Model

Figure 9 is an ROC Curve generated for the model. The Receiver Operating Characteristic (ROC) curve is a graphical plot used to assess the performance of a binary classification model. It illustrates the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) at various threshold settings. The area under the curve is 0.99, which is close to 1, indicating excellent classification performance.

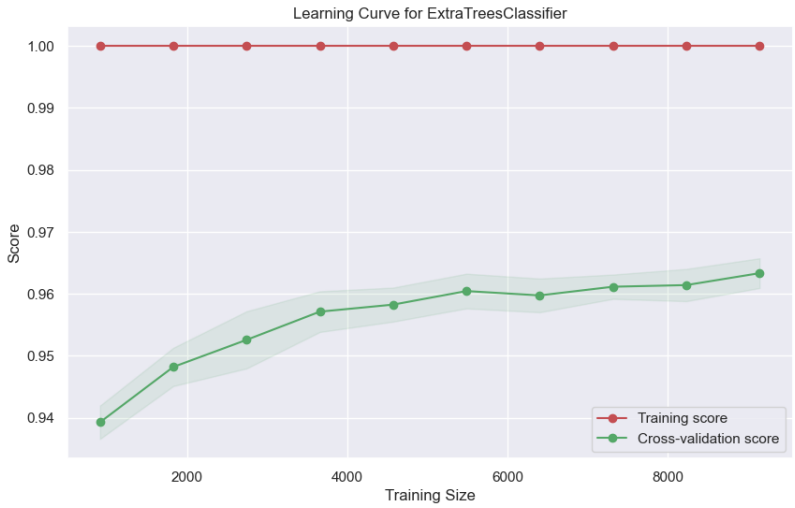

Figure 10: Learning Curve of Extra Trees Model

Figure 10 is a learning curve which indicates the performance of the model in the training phase and validation phase. The learning curve shows how the model performs on the training set and validation set as the size of the dataset changes. The validation score is near the training score, which shows that the model performs well even with new data.

7. Conclusion

Phishing detection is an important feature of cybersecurity, especially given the increasing amount of assaults that target naïve people via malicious URLs. Phishing websites are becoming increasingly sophisticated, necessitating the use of artificial intelligence (AI) to detect and prevent assaults. The development of an AI-based phishing detection system necessitates the collection of a dataset that includes the characteristics of URLs linked with phishing and authentic sites. To overcome this issue, the ExtraTrees Classifier is presented in this work. The model is trained using a dataset that includes URL length, domain information, special character counts, and other phishing-related variables. The model is trained using a training dataset; finally,

the proposed approach was evaluated in steps with accuracy, precision, recall and f1-score. The ExtraTrees Classifier achieved impressive results (accuracy = 98.75%,precision = 97.5%, recall = 95% and F1-score = 96.2%). Phishing assaults are a major problem due to their ever-changing nature. Current machine learning algorithms for phishing detection frequently fail to adapt to new cybercriminal techniques. As a result, there is an urgent need to create better-trained models that can successfully detect and neutralize phishing threats.

Acknowledgement: The support of all my co-authors is highly appreciated.

Data Availability Statement: The study makes use of a dataset that contains URL-related attributes, including traits that indicate phishing activity.

Funding Statement: No funding has been obtained to help prepare this manuscript and research work.

Conflicts of Interest Statement: No conflicts of interest have been declared by the authors. Citations and references are mentioned in the information used.

Ethics and Consent Statement: The consent was obtained from the organization and individual participants during data collection, and ethical approval and participant consent were received.

References

1. A. Karim, M. Shahroz, K. Mustofa, S. B. Belhaouari and S. R. K. Joga, "Phishing Detection System Through Hybrid Machine Learning Based on URL," in IEEE Access, vol. 11, no.3, pp. 36805-36822, 2023.
2. W. Ali and A. A. Ahmed, "Hybrid intelligent phishing website prediction using deep neural networks with genetic algorithm-based feature selection and weighting," IET Information Security, vol. 13, no. 6, pp. 659-669, 2019
3. D. Jibat, S. Jamjoom, Q. A. Al-Haija and A. Qusef, "A Systematic Review: Detecting Phishing Websites Using Data Mining Models," in Intelligent and Converged Networks, vol. 4, no. 4, pp. 326-341, 2023.
4. E. Zhu, Z. Chen, J. Cui and H. Zhong, "MOE/RF: A Novel Phishing Detection Model Based on Revised Multi-Objective Evolution Optimization Algorithm and Random Forest," in IEEE Transactions on Network and Service Management, vol. 19, no. 3, pp. 4461-4478, 2022.
5. I. Kara, M. Ok and A. Ozaday, "Characteristics of Understanding URLs and Domain Names Features: The Detection of Phishing Websites With Machine Learning Methods," in IEEE Access, vol. 10, no.11, pp. 124420-124428, 2022.
6. L. R. Kalabarige, R. S. Rao, A. R. Pais and L. A. Gabralla, "A Boosting-Based Hybrid Feature Selection and Multi-Layer Stacked Ensemble Learning Model to Detect Phishing Websites," in IEEE Access, vol. 11, no.7, pp. 71180-71193, 2023.
7. L. Tang and Q. H. Mahmoud, "A Deep Learning-Based Framework for Phishing Website Detection," in IEEE Access, vol. 10, no.12, pp. 1509-1521, 2022.
8. P. Dhinakaran, M.A. Thinesh, and M. Paslavskyi, "Enhancing Cyber Intrusion Detection through Ensemble Learning: A Comparison of Bagging and Stacking Classifiers," FMDB Transactions on Sustainable Computer Letters., vol. 1, no. 4, pp.210 –227, 2023.
9. P. Yang, G. Zhao and P. Zeng, "Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning," in IEEE Access, vol. 7,no.1, pp. 15196-15209, 2019.
10. R. Zieni, L. Massari and M. C. Calzarossa, "Phishing or Not Phishing? A Survey on the Detection of Phishing Websites," in IEEE Access, vol. 11, no.2, pp. 18499-18519, 2023, doi: 10.1109/ACCESS.2023.3247135.
11. S. Al-Ahmadi, A. Alotaibi and O. Alsaleh, "PDGAN: Phishing Detection With Generative Adversarial Networks," in IEEE Access, vol. 10, no. 4, pp. 42459-42468, 2022.
12. S. Mohanty, A. A. Acharya, T. Gaber, N. Panda, E. Eldesouky and I. A. Hameed, "An Efficient Hybrid Feature Selection Technique Toward Prediction of Suspicious URLs in IoT Environment," in IEEE Access, vol. 12, no. 4, pp. 50578-50594, 2024.
13. Y. A. Alsariera, V. E. Adeyemo, A. O. Balogun and A. K. Alazzawi, "AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites," in IEEE Access, vol. 8, no. 8, pp. 142532-142542, 2020,
14. Y. Wei and Y. Sekiya, "Sufficiency of Ensemble Machine Learning Methods for Phishing Websites Detection," in IEEE Access, vol. 10, no.11, pp. 124103-124113, 2022.
15. A. Ghosh, A. Banerjee, and S. Das, "Design of compact polarization insensitive triple band stop frequency selective surface with high stability under oblique incidence," Radioengineering, vol. 28, no. 3, pp. 552–558, 2019.
16. A. Ghosh, A. Ghosh, and J. Kumar, "Circularly polarized wide-band quad-element MIMO antenna with improved axial ratio bandwidth and mutual coupling," IEEE Antennas Wireless Propag. Lett., vol. 23, no. 12, pp. 4718-4722, 2024.

17. A. Ghosh, A. Mitra, and S. Das, “Meander line-based low profile RIS with defected ground and its use in patch antenna miniaturization for wireless applications,” Microwave Opt. Technol. Lett., vol. 59, no. 3, pp. 732–738, 2017.
18. A. Ghosh, T. Mandal, and S. Das, “Design and analysis of annular ring-based RIS and its use in dual-band patch antenna miniaturization for wireless applications,” J. Electromagn. Waves Appl., vol. 31, no. 3, pp. 335–349, 2017.
19. A. Gupta, N. Mahesh, S. K. Bairappaka, and A. Ghosh, “Comparison of the performance of L and Pi matching networks for design of a 2.4 GHz RF-DC rectifier for RF energy harvesting,” in Proc. 2024 IEEE 4th Int. Conf. Sustainable Energy and Future Electric Transportation (SEFET), Hyderabad, India, pp. 1–5, 2024.
20. A. Kulkarni, “Image Recognition and Processing in SAP HANA Using Deep Learning,” International Journal of Research and Review Techniques, vol. 2, no. 4, pp. 50-58, 2024.
21. A. Kulkarni, “Supply Chain Optimization Using AI and SAP HANA: A Review,” International Journal of Research Radicals in Multidisciplinary Fields, vol. 2, no. 2, pp. 51-57, 2024.
22. A. R. Neravetla, V. K. Nomula, A. S. Mohammed, and S. Dhanasekaran, “Implementing AI-driven Diagnostic Decision Support Systems for Smart Healthcare,” in 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), Mandi, Himachal Pradesh, India, pp. 1–6, 2024.
23. A. S. Mohammed, A. R. Neravetla, V. K. Nomula, K. Gupta, and S. Dhanasekaran, “Understanding the Impact of AI-driven Clinical Decision Support Systems,” in 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), Mandi, Himachal Pradesh, India, pp. 1–6, 2024.
24. A. Thirunagalingam, S. Addanki, V. R. Vemula, and P. Selvakumar, “AI in Performance Management: Data-Driven Approaches,” in Advances in Business Strategy and Competitive Advantage, IGI Global, USA, pp. 101–126, 2024.
25. A. Virmani and M. Kuppam, “Designing fault-tolerant modern data engineering solutions with reliability theory as the driving force,” in 2024 9th International Conference on Machine Learning Technologies (ICMLT), Oslo, Norway, pp. 265–272, 2024.
26. A. Virmani and M. Kuppam, “MLOps Antipatterns and Mitigation Approaches,” Int. J. Comput. Trends Technol., vol. 72, no. 2, pp. 9–15, 2024.
27. G. S. Sahoo and A. Ghosh, “Performance analysis for hybrid beamforming algorithm in 5G MIMO wireless communication system,” in Proc. 2022 IEEE Microwaves, Antennas, and Propagation Conf. (MAPCON), Bangalore, India, pp. 592–596, 2022.
28. H. Mistri, A. Ghosh, A. R. Sardar, and B. Choudhury, “Performance enhancement of graphene-based linear to circular polarization converter for terahertz frequency using a novel parameter prediction methodology,” Plasmonics, pp. 1-15, 2024, Press.
29. H. Mistri, A. Ghosh, and M. Dandapathak, “Bidirectional triple-band truly incident angle insensitive polarization converter using graphene-based transmissive metasurface for terahertz frequency,” Frequenz, vol. 78, no. 11-12, pp. 569-579, 2024.
30. K. Mazumder and A. Ghosh, “A small scale circular polarized reader antenna with wide beamwidth for RFID applications,” in Proc. 2022 IEEE Wireless Antenna and Microwave Symp. (WAMS), Rourkela, India, pp. 1–5, 2022.
31. K. Mazumder, A. Ghosh, A. Bhattacharya, S. Ahmad, A. Ghaffar, and M. Hussein, “Frequency switchable global RFID tag antennae with metal compatibility for worldwide vehicle transportation,” Sensors, vol. 23, no. 8, p. 3854, 2023.
32. K. Oku, L. S. Samayamantri, S. Singhal, and R. Steffi, “Decoding AI decisions on depth map analysis for enhanced interpretability,” in Advances in Computer and Electrical Engineering, IGI Global, USA, pp. 143–164, 2024.
33. K. Oku, R. K. Vaddy, A. Yada, and R. K. Batchu, “Data Engineering Excellence: A Catalyst for Advanced Data Analytics in Modern Organizations,” International Journal of Creative Research in Computer Technology and Design, vol. 6, no. 6, pp. 1–10, 2024.
34. L. S. Samayamantri, S. Singhal, O. Krishnamurthy, and R. Regin, “AI-driven multimodal approaches to human behavior analysis,” in Advances in Computer and Electrical Engineering, IGI Global, USA, pp. 485–506, 2024.
35. L. Thammareddi, M. Kuppam, K. Patel, D. Marupaka, and A. Bhanushali, “An extensive examination of the DevOps pipelines and insightful exploration,” Int. J. Comput. Eng. Technol., vol. 14, no. 3, pp. 76–90, 2023.
36. M. Kuppam, “Enhancing reliability in software development and operations,” Int. Trans. Artif. Intell., vol. 6, no. 6, pp. 1–23, 2022.
37. M. Kuppam, “Observability practice with OODA principles and processes,” Sch. J. Eng. Tech., vol. 11, no. 11, pp. 302–308, 2023.
38. M. Kuppam, M. Godbole, T. R. Bammidi, S. S. Rajest, and R. Regin, “Exploring innovative metrics to benchmark and ensure robustness in AI systems,” in Explainable AI Applications for Human Behavior Analysis, IGI Global, USA, pp. 1–17, 2024.
39. M. Midya, A. Ghosh, and M. Mitra, “Meander-line-loaded circularly polarized square-slot antenna with inverted-L-shaped feed line for C-band applications,” IET Microwaves, Antennas & Propag., vol. 15, no. 11, pp. 1425–1431, 2021.

40. M. Parveen Roja, M. Kuppam, S. K. R. Koduru, R. Byloppilly, S. D. Trivedi, and S. S. Rajest, “An aid of business intelligence in retailing services and experience using artificial intelligence,” in Cross-Industry AI Applications, IGI Global, USA, pp. 14–30, 2024.
41. P. Pulivarthy, “Enhancing Dynamic Behaviour in Vehicular Ad Hoc Networks through Game Theory and Machine Learning for Reliable Routing,” International Journal of Machine Learning and Artificial Intelligence, vol. 4, no. 4, pp. 1–13, 2023.
42. P. Pulivarthy, “Performance Tuning: AI Analyse Historical Performance Data, Identify Patterns, and Predict Future Resource Needs,” International Journal of Innovations in Applied Sciences and Engineering, vol. 8, no. 2, pp. 139–155, 2022.
43. P. S. Venkateswaran, F. T. M. Ayasrah, V. K. Nomula, P. Paramasivan, P. Anand, and K. Bogeshwaran, “Applications of artificial intelligence tools in higher education,” in Advances in Business Information Systems and Analytics, USA: IGI Global, pp. 124–136, 2023.
44. R. C. Komperla, K. S. Pokkuluri, V. K. Nomula, G. U. Gowri, S. S. Rajest, and J. Rahila, “Revolutionizing Biometrics with AI-Enhanced X-Ray and MRI Analysis,” in Advancements in Clinical Medicine, P. Paramasivan, S. Rajest, K. Chinnusamy, R. Regin, and F. J. Joseph, Eds. USA: IGI Global, pp. 1–16, 2024.
45. R. S. Gaayathri, S. S. Rajest, V. K. Nomula, and R. Regin, “Bud-D: enabling bidirectional communication with ChatGPT by adding listening and speaking capabilities,” FMDB Transactions on Sustainable Computer Letters, vol. 1, no. 1, pp. 49–63, 2023.
46. S. Chundru, “Ensuring Data Integrity Through Robustness and Explainability in AI Models,” Transactions on Latest Trends in Artificial Intelligence, vol. 1, no. 1, pp. 1-19, 2020.
47. S. Chundru, “Leveraging AI for Data Provenance: Enhancing Tracking and Verification of Data Lineage in FATE Assessment,” International Journal of Inventions in Engineering & Science Technology, vol. 7, no.1, pp. 87-104, 2021.
48. S. Genikala, A. Ghosh, and B. Roy, “Triple band single layer microwave absorber based on closed loop resonator structures with high stability under oblique incidence,” AEU-Int. J. Electron. Commun., vol. 164, no.5, no. 154629, 2023.
49. S. K. Bairappaka and A. Ghosh, “Co-planar waveguide fed dual band circular polarized slot antenna,” in Proc. 2020 3rd Int. Conf. Multimedia Process. Commun. Inf. Technol. (MPCIT), Shivamogga, India, pp. 10–13, 2020.
50. S. K. Bairappaka, A. Ghosh, O. Kaiwartya, A. Mohammad, Y. Cao, and R. Kharel, “A novel design of broadband circularly polarized rectenna with enhanced gain for energy harvesting,” IEEE Access, vol. 12, no.5, pp. 65583–65594, 2024.
51. S. Nej and A. Ghosh, “Quad elements dual band MIMO antenna for advanced 5G technology,” in Proc. 2020 IEEE 4th Conf. Inf. Commun. Technol. (CICT), Chennai, India, pp. 1–5, 2020.
52. S. Nej, S. K. Bairappaka, B. N. V. Sai Durga Sri Raja Ram Dinavahi, S. Jana, and A. Ghosh, “Design of a high order dual band MIMO antenna with improved isolation and gain for wireless communications,” Arab. J. Sci. Eng., pp. 1–18, 2024, Press.
53. S. S. Ramesh, A. Jose, P. R. Samraysh, H. Mulabagala, M. S. Minu, and V. K. Nomula, “Domain Generalization and Multidimensional Approach for Brain MRI Segmentation Using Contrastive Representation Transfer Learning Algorithm,” in Advancements in Clinical Medicine, P. Paramasivan, S. Rajest, K. Chinnusamy, R. Regin, and F. J. Joseph, Eds. USA: IGI Global, pp. 17–33, 2024.
54. V. K. Nomula, A. S. Mohammed, A. R. Neravetla, and S. Dhanasekaran, “Leveraging Deep Learning in Implementing Efficient Healthcare Processes,” in 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), Mandi, Himachal Pradesh, India, pp. 1–6, 2024.
55. V. R. Vemula, “Adaptive threat detection in DevOps: Leveraging machine learning for real-time security monitoring,” Int. Mach. Learn. J. Comput. Eng., vol. 5, no. 5, pp. 1–17, 2022.
56. V. R. Vemula, “Recent Advancements in Cloud Security Using Performance Technologies and Techniques,” 2023 9th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, pp. 1-7, 2023.
57. V. S. K. Settibathini, A. Virmani, M. Kuppam, Nithya, S. Manikandan, and Elayaraja, “Shedding light on dataset influence for more transparent machine learning,” in Explainable AI Applications for Human Behavior Analysis, IGI Global, USA, pp. 33–48, 2024.