Protection Against Spear Phishing Attacks Using the Ensemble Method of Machine Learning

Jecinta Ifechukwu Fidelis
Student ID: 23148306

School of Computing
National College of Ireland

Supervisor:Imran Khan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Jecinta Ifechukwu Fidelis ……. ……

Student ID: 23148306 ………..……

Programme: Msc in Cyber Security ……………………………………………………………… **Year:** 2024 …………………………..

Module: Msc Research Practicum …….………

Lecturer: Imran Khan …….………

Submission Due Date: 16/09/2024 …….………

Project Title: …….………

Word Count: ……………………………………… **Page Count:** …………………………………….…….………

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Jecinta Ifechukwu Fidelis ………

Date: 16/09/2024 ………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Protection Against Spear Phishing Attacks Using the Ensemble Method of Machine Learning

Jecinta Ifechukwu Fidelis
23148306

Abstract

This study investigated and implemented the use of multiple ensemble machine learning methods to enhance the detection and classification of spear phishing emails, which is a critical challenge in cybersecurity due to the sophisticated nature of phishing attacks. The primary motivation for this study is the need for more effective phishing detection systems that can accurately distinguish between legitimate emails and spear phishing attempts beyond the human eyes. To address this issue, the study focused on the following three key objectives: enhancing feature extraction techniques, implementing ensemble machine learning models, and deploying a practical phishing detection system. The research employed advanced text feature extraction methods with specific regard to Term Frequency-Inverse Document Frequency (TF-IDF), to convert email content into numerical vectors, thereby improving model accuracy. In this study, LightGBM emerged as the most effective model in test experiments and outperforming traditional models like Logistic Regression and Naive Bayes. To ensure practical applicability, this model was deployed with a user-friendly interface developed using Gradio, enabling real-time email classification. This integration provides a practical solution for organisations to combat spear phishing attacks. The study's findings demonstrate the efficacy of ensemble models in improving phishing detection and offer significant implications for both academic research and practical applications. Future work will explore adaptive learning approaches to further enhance the system's resilience against evolving phishing tactics and address limitations identified in dynamic environments.

Key words: Ensemble Machine Learning, Spear Phishing Detection, Email Classification, LightGBM, Cybersecurity

1 Introduction

Phishing is a cyber-attack method by which criminals employ the use of deceptive means such as fraudulent and curated emails or illegitimate websites to trick recipients into revealing sensitive information such as credit card numbers, passwords and usernames (Alshingiti et al., 2023). The criminals involved rely on casting a wide net, such as by sending out large number of emails with the hope that a small percentage of recipients will fall for the scam (Goenka, Chawla and Tiwari, 2023). Based on the law of large numbers, a few number of victim usually fall victim. The Law of Large Numbers (Sternstein, 2024) and the concept of Spray and Pray (Wrightson, 2015), suggest in this context that when criminals send out a large volume of fraudulent messages such as phishing emails, the expectation that at least a few susceptible people will fall victim. Mathematically, this can also be explained through the law of probability in which chance increases with more trials. The problem is further compounded by the fact that phishing messaging often contains generic greetings with common themes like account verification, lottery winnings, or urgent security updates etc., strategies that are sometimes used by legitimate businesses. This increases the difficult for especially the average person to be able to differentiate between normal and spoofed emails.

There are many types of phishing attacks. Some of these include email phishing, whale phishing, spear phishing, website phishing, smishing (sms phishing) and vishing (voice phishing) (Rebovich and Byrne, 2022). The many phishing types fall under social engineering-based phishing, DNS-based phishing, content injection phishing, DHCP-based phishing, proxy-based phishing, search engine phishing, and man-in-the-middle attack (Sonowal, 2021). The scope of this section does not permit exploring all the many types of phishing given that the study pertains to spear phishing in the social engineering category. Spear phishing attacks are unique and worth to be studied because they are highly targeted and carefully curated at individuals (Thakur et al., 2023). Attackers can go to the extent of gathering detailed information about their victims from social media profiles, public records, and other online sources. This information is then used to create emails that will appear to come from trusted colleagues, business partners, or familiar organisations. The emails usually align closely with victims' circumstances, expectations and experiences. Conse Alkhalil et al. (2021) quently, the chance for success can significantly higher compared to other types, which is why Alkhalil et al. (2021) highlighted that it is drawing the attention of more phishers in recent times.

Spear phishing has proven to be a particularly devastating form of phishing attack as indicated by several high-profile cases that highlight its impact across various sectors. In 2020 for example, Twitter experienced a significant breach when attackers targeted employees through a spear phishing campaign (BBC, 2020). This enabled the attackers to gain access to internal systems and hijack the accounts of prominent figures like Elon Musk and Barack Obama, which they subsequently used to promote a cryptocurrency scam. The incident not only resulted in financial losses but also severely damaged Twitter's reputation and led to increased scrutiny of its security practices. Another case study in the financial sector involved Robinhood, a financial trading platform in 2021, where a customer support employee was tricked into providing access to internal systems that comprised the data of about seven million customers (BBC, 2021). Governments are not also speared as some MPs in the UK recently faced spear phishing attacks that attempted to compromised the parliament (Quinn and Courea, 2024). These cases collectively illustrate the importance of studying spear phishing and the significant impact that spear phishing can have, from financial losses and reputational damage to national security concerns and the compromise of sensitive data.

Just like it is difficult for the natural human eye to detect AI writing, warranting the use of emerging AI detectors, it can also be difficult for a human being to detect spear phishing emails with their natural eyes. However, current machine-enabled detection methods have some limitations (Goenka, Chawla and Tiwari, 2023). These conventional tools are not designed to analyse the contextual details and personalized elements that characterize spear phishing attempts (Thakur et al., 2023). Moreover, there is the challenge of high false-positive rates (Alnemari and Alshammari, 2023). Furthermore, without a feedback mechanism from the user, there may not be room for them to adapt quickly enough to evolving phishing tactics These limitations highlight the necessity for continuous research and development of more sophisticated and adaptive detection models.

The potential benefits of improving spear phishing detection include enhanced detection capabilities that would significantly bolster cybersecurity, protecting sensitive information and maintaining the integrity of digital communications (Alnemari and Alshammari, 2023). Organisations can avoid the substantial financial costs associated with incident response, legal ramifications, and loss of consumer trust by reducing the risk of data breaches. As noted by Alkhalil et al. (2021), organisations that implement robust spear phishing defences are better positioned to safeguard their assets and reputation, ensuring continuity of operations and customer trust. Enhanced detection methods can contribute to the overall resilience of the digital ecosystem, reducing the incidence of cybercrime and fostering a safer online environment (Tzavara and Vassiliadis, 2024). This study aims to make a significant contribution to cybersecurity practices by contributing to current defences and implementing ensemble machine learning techniques to combat evolving phishing tactics. A key contribution is to provide a two-way flag feedback mechanism where man and machine will work together to provide a robust defence against spear phishing. This is unlike conventional methods where usually only the machine makes the decisions.

The primary research question guiding this study is:

How can ensemble machine learning methods be utilized to improve the detection and classification of spear phishing emails?

This question aims to explore the application and effectiveness of ensemble machine learning techniques in identifying and classifying spear phishing emails accurately. By leveraging multiple models and combining their strengths, the study seeks to enhance the overall detection performance and provide a robust solution to the problem of spear phishing.

To address the primary research question, the study has outlined the following specific research objectives:

1. Enhance Feature Extraction Techniques for Improved Detection Accuracy:
    - To implement and refine text feature extraction methods, specifically Term Frequency-Inverse Document Frequency (Tf-idf). This objective focuses on converting email content into numerical vectors that can be effectively used by machine learning models, thereby improving their ability to distinguish between phishing and non-phishing emails.
2. Ensemble Machine Learning Models:
    - To implement and test spear phishing detection models leveraging ensemble machine learning methods. The goal is to assess ensemble models against various single machine learning models. The aim is to identify the most

accurate and efficient model for phishing email detection, using evaluation metrics such as accuracy, precision, recall, and F1-score.

3. Implement and Validate a Phishing Detection Model Deployment:

    - To deploy the best-performing machine learning model to provide a functional interface that allows users to efficiently classify emails as either phishing or normal. This objective ensures that the solution is practical and user-friendly, enabling quick and accurate determination of email legitimacy. The deployment will involve creating an interface that integrates the model into a real-world application scenario.

2 Related Work

2.1 Spear phishing

Over the past decade, spear phishing has evolved into an increasingly prevalent and sophisticated threat. According to Halevi, Memon, and Nov (2015), the targeted nature of spear phishing distinguishes it from other forms of phishing by focusing on specific individuals or organizations. This precision makes spear phishing especially dangerous, as the attacker typically gathers detailed information about the target, often through social media and professional networking sites, to craft a highly convincing and personalized message. This method has proven effective, as demonstrated by the FBI's Internet Crime Complaint Center (IC3) report, which identified phishing and related scams as the most common type of cybercrime in 2020, with losses exceeding $1.8 billion (Internet Crime Complaint Center, 2020).

The rise in spear phishing can be attributed to several factors. The widespread use of social media and professional networking sites, such as LinkedIn and Facebook, has made it easier for attackers to gather detailed information about potential targets (Basit et al., 2020; Bossetta, 2018). This readily available data allows attackers to craft emails that are not only personalized but also highly relevant to the target's professional or personal life. In addition to social engineering, attackers are continually evolving their tactics by using more convincing language, mimicking legitimate email formats, and employing advanced techniques such as domain spoofing and business email compromise (Suzuki and Monroy, 2021). Moreover, the use of automation and artificial intelligence tools has enabled attackers to scale their spear phishing campaigns, creating highly personalized emails more efficiently and targeting a larger number of individuals.

Recent trends indicate that the use of artificial intelligence (AI) and machine learning (ML) by attackers is on the rise, allowing them to analyze vast amounts of data to identify potential targets and tailor their attacks accordingly. AI can also be used to generate convincing phishing emails that are indistinguishable from legitimate communications, further increasing the success rate of these attacks (Qi et al., 2023). Additionally, the integration of AI in spear phishing attacks has led to the emergence of more sophisticated tactics, such as deepfake phishing, where attackers use AI-generated audio or video to impersonate executives or other high-profile individuals, making it even more challenging for victims to detect the scam.

The consequences of successful spear phishing attacks are severe and far-reaching, affecting both individuals and organizations. Financial losses are one of the most immediate impacts, as victims may be tricked into transferring funds to fraudulent accounts or revealing their banking credentials (Shevchenko et al., 2023). For businesses, these losses can be compounded by the costs associated with fraudulent transactions, legal fees, fines, and the recovery of stolen funds. Beyond immediate financial losses, spear phishing is often used as an entry point for more extensive cyber-attacks, leading to data breaches. Once attackers gain access to an individual's credentials, they can infiltrate corporate networks, steal sensitive data, and compromise confidential information. Data breaches resulting from spear phishing can expose personal information, intellectual property, and trade secrets, leading to severe operational and legal repercussions (Merz, Fallon, and Scalco, 2018).

Organizations targeted by spear phishing attacks may also experience lasting reputational damage. Customers and clients may lose trust in a company's ability to protect their information, leading to loss of business and a damaged brand image. High-profile breaches can attract negative media attention, further eroding public confidence and making it difficult for the organization to recover (Li, Xiao, and Zhang, 2023). In addition to reputational damage, spear phishing attacks can disrupt normal business operations. Organizations may need to divert resources to manage the breach, investigate the incident, and implement additional security measures, which can lead to decreased productivity and operational inefficiencies.

2.2 Critical Evaluation of Recent Trends and Techniques in Spear Phishing Detection

Recent advancements in spear phishing detection have undeniably made significant strides in enhancing cybersecurity defenses. However, these developments also present limitations and challenges that must be critically assessed to avoid over-reliance on technology or strategies that may not be as effective in practice as they appear in theory.

2.2.1 Artificial Intelligence and Machine Learning

The integration of artificial intelligence (AI) and machine learning (ML) into spear phishing detection frameworks is often lauded as a major breakthrough (Basit et al., 2020; Jackson, 2023). AI and ML are generally able to process vast datasets, identify patterns, and adapt to new threats with remarkable speed. However, this reliance on AI-driven systems is not without its challenges. For instance, AI models are only as good as the data they are trained on (Budach et al., 2022). Consequently, if the training data is not sufficiently diverse or representative of real-world scenarios, these models may fail to detect novel or sophisticated spear phishing attacks. Moreover, AI and ML models are vulnerable to adversarial attacks, where attackers deliberately manipulate input data to fool the model (Lin et al., 2021). For example, slight alterations to an email's content or structure could bypass detection algorithms, rendering AI-driven systems less effective. Additionally, the complexity and opacity of some machine learning models can make it difficult for cybersecurity professionals to understand how these systems arrive at their conclusions, potentially leading to blind spots in threat detection (Frasca et al., 2024). This black-box nature of AI models raises concerns about their reliability and the potential for attackers to exploit these systems.

2.2.2 Behavioural Analytics: A Double-Edged Sword

Behavioural analytics and user profiling are increasingly used to detect deviations from normal user behaviour that could indicate a spear phishing attempt (Basit et al., 2020). While these techniques can be highly effective in identifying anomalies, they also come with significant privacy concerns and operational challenges. Monitoring user behaviour at a granular level requires access to vast amounts of personal and sensitive data, raising ethical questions about surveillance and data privacy. Furthermore, the effectiveness of behavioural analytics is heavily dependent on the quality of the baseline data (Bruhn et al., 2018). In dynamic and rapidly changing work environments, what constitutes normal behaviour can shift frequently, leading to potential false positives or, conversely, missed detections. For example, a sudden increase in remote work due to global events like the COVID-19 pandemic could alter user behaviour patterns, confusing analytics systems and leading to incorrect assessments of what is considered anomalous.

2.2.3 Multi-Factor Authentication: Not a Panacea

Multi-factor authentication (MFA) has long been touted as a key defense against spear phishing, adding a crucial layer of security by requiring additional verification steps beyond just a password (Ogbanufe and Baham, 2022). However, MFA is not infallible. Attackers have developed sophisticated methods to bypass MFA, such as man-in-the-middle attacks, where the attacker intercepts and relays authentication codes, or SIM-swapping, where an attacker takes control of a victim's phone number to receive authentication codes. In addition, the implementation of MFA can create friction for users, leading to decreased productivity and resistance from employees, particularly in environments where quick access to systems is critical (Plascencia, Díaz–Damacillo and Robles-Agudo, 2020). This usability challenge can result in organizations opting for weaker or more convenient MFA methods, thereby undermining the security benefits MFA is supposed to provide.

2.2.4 Real-Time Detection: The Challenge of Speed vs. Accuracy

Real-time phishing detection and response systems promise to neutralize threats as they occur, potentially stopping spear phishing attacks before they cause significant harm (Basit et al., 2020). However, the balance between speed and accuracy in these systems is delicate. Real-time systems must process vast amounts of data quickly, which can lead to a trade-off between the thoroughness of the analysis and the need for immediate action. Moreover, real-time systems often rely on automated responses, such as quarantining emails or blocking accounts. While these actions can prevent the spread of an attack, they can also disrupt legitimate business operations if triggered by false positives (Monge and Soriano, 2023). For instance, a legitimate but unusual transaction or communication might be flagged as suspicious, leading to unnecessary delays or operational disruptions. Therefore, the reliance on real-time systems also raises concerns about the scalability and sustainability of these solutions in large organizations with complex infrastructures.

2.2.5 Advanced Social Engineering Detection: Limitations and Human Factors

Detecting the social engineering components of spear phishing attacks remains a significant challenge (Mashtalyar et al., 2021). While AI and psychological profiling techniques have made progress in identifying manipulative language and tactics, these systems are far from foolproof. Social engineering is inherently human-centered, exploiting emotions, cognitive biases, and trust. As such, purely technical solutions may not fully capture the subtleties of human communication that skilled attackers can manipulate. Furthermore, the effectiveness of social engineering detection systems is limited by the diversity and creativity of the attackers. As these systems become more advanced, attackers are likely to adapt by crafting even more sophisticated and nuanced phishing attempts that evade detection. This cat-and-mouse dynamic highlights the ongoing challenge of relying on technology alone to combat human-centric threats.

2.2.6 Zero-Trust Architecture and Network Segmentation: Implementation Challenges

Zero-trust architecture is increasingly advocated as a robust defense against spear phishing, operating on the principle that no user or device should be trusted by default. While this approach offers significant security benefits, its implementation is not without challenges. Adopting a zero-trust model requires a fundamental shift in how organizations manage access and identity, which can be complex and resource-intensive. Moreover, the effectiveness of zero-trust depends on the continuous monitoring and verification of all users and devices, which can strain IT resources and infrastructure. In practice, maintaining a zero-trust environment can be difficult, especially in large organizations with diverse and distributed networks. Additionally, the human factor plays a critical role; if users find zero-trust protocols cumbersome, they may seek workarounds that undermine the system’s integrity.

2.3 Review of Studies Related to Spear Phishing

A comprehensive study by Alkhalil et al. (2021) categorizes technical solutions for phishing into detection, prevention, corrective measures, warning tools, and authentication techniques. This framework establishes the necessity of a multi-layered defense strategy, which is particularly important given the sophistication of modern spear phishing attacks. However, traditional machine learning (ML) techniques have shown limitations, especially against zero-day attacks, where attackers use previously unseen tactics. Evans et al. (2021) addressed this issue by proposing a reinforcement learning-based model called RAIDER, which autonomously identifies relevant features, reduces feature dimensions, and enhances detection accuracy. While RAIDER represents a significant advancement over static methods, its computational complexity poses practical deployment challenges, especially for smaller organizations with limited resources.

Bossetta (2018) shifts the focus from traditional email-based attacks to the exploitation of social media platforms by state-sponsored groups. Their five-phase model—Collect, Construct, Contact, Compromise, Contagion—emphasizes the sophistication of these campaigns and the need for detection methods that account for the social engineering aspects of spear phishing. This model complements the technical solutions discussed by Alkhalil et al. (2021), highlighting the necessity of adapting machine learning techniques to newer attack vectors beyond traditional email. The increasing use of social media for spear phishing, particularly by state-sponsored actors, underscores the need for more sophisticated detection tools that can identify and mitigate these evolving threats.

The works of McConnell et al. (2023) and Chandra et al. (2019) highlight the importance of ensemble learning methods in phishing detection. McConnell et al. (2023) emphasized the role of hyperparameter tuning and feature selection in optimizing machine learning models, achieving high performance with ensemble methods like boosting, bagging, stacking, and voting. Ensemble methods have proven effective in general phishing detection and show promise for application to spear phishing, although their direct relevance to targeted attacks remains a subject of ongoing research. Similarly, Chandra et al. (2019) demonstrated the efficacy of ensemble algorithms in phishing detection, suggesting their potential applicability to spear-phishing detection. However, their focus on general phishing filtering limits the direct relevance of their findings to the specific challenges posed by spear phishing.

Qi et al. (2023) introduced novel ensemble methods (FMPED and FMMPED) that enhance phishing email detection by creating a new training set through undersampling. These methods achieve impressive performance but rely heavily on specific dataset characteristics, which may limit their generalizability across diverse phishing scenarios. This highlights the ongoing challenge of developing universally robust machine learning models capable of adapting to the dynamic nature of phishing attacks. This theme is consistent with the findings of Evans et al. (2021), who emphasized the adaptability of machine learning techniques as a critical factor in improving detection accuracy for spear phishing.

Furthermore, Li and Cheng (2023) proposed a few-shot learning method for scenarios with limited training data, leveraging word-embedding techniques to optimize email content features. This approach demonstrated superior performance in small sample sizes, contrasting with the more resource-intensive methods discussed by Evans et al. (2021) and McConnell et al. (2023). However, the simplicity of the proposal by Li and Cheng (2023) may struggle against more sophisticated spear-phishing tactics, particularly those that involve advanced social engineering or AI-generated content. This highlights the ongoing trade-off between model simplicity and detection accuracy, a critical consideration in the development of effective spear phishing detection methods.

2.4 Gap and Motivation

Recent studies on spear phishing have provided valuable insights into the evolving tactics used by attackers and the effectiveness of various defense mechanisms. While advancements in machine learning, behavioural analytics, and zero-trust architecture offer promising approaches to detecting and mitigating spear phishing, these methods also have limitations. Psychological manipulation remains a significant challenge, requiring more sophisticated training programs tailored to individual susceptibilities. Machine learning models, though powerful, depend heavily on the quality and diversity of their training data, which can limit their effectiveness against novel attacks. To address these limitations, there is an increasing recognition that solutions must not be solely machine-driven; instead, they should incorporate a backward communication loop, or feedback mechanism, where human input and machine learning systems continuously interact and improve together. This collaborative approach allows for the refinement of detection algorithms based on real-time human feedback, increasing detection precision.

This study is motivated by the critical need to enhance the detection and prevention of such attacks, given their growing prevalence and the severe consequences they entail. Spear phishing incidents have been on a sharp rise in recent years, reflecting the increasing vulnerability of both personal and corporate data to such attacks as highlighted by Alkhalil et al. (2021). It has become one of the most effective methods for cybercriminals to breach security defenses, with incidents rising dramatically. The sophistication of these attacks has grown, with perpetrators leveraging advanced social engineering techniques and extensive personal data obtained from social media and other online sources to craft highly personalized phishing emails (Bossetta, 2018). This increasing sophistication, coupled with the integration of AI and ML techniques by attackers, presents significant challenges for existing detection methods. As a result, there is a pressing need for continued research and development in this area to keep pace with the evolving tactics of cybercriminals.

3 Research Methodology

This section details the comprehensive methodology employed to investigate the research question. It describes the research design, data collection methods, data preprocessing, feature extraction, model implementation, evaluation metrics, and the deployment of the phishing detection model. The methodology is informed by Kamiri and Mariga (2021), ensuring a systematic and scientific approach to achieving the research objectives.

3.1 Research Design

The study adopts a quantitative research design, leveraging machine learning algorithms to analyze and classify email data. This approach is particularly suitable for the nature of the research, as it allows for the processing of large volumes of data and the application of sophisticated statistical methods to detect patterns indicative of phishing attempts. The research follows a structured procedure from data collection to model deployment, ensuring reproducibility and reliability. By utilizing ensemble learning methods, the study aims to combine the strengths of multiple models to enhance the accuracy and robustness of the phishing detection system.

3.2 Data Transformation and Pre-processing

The effectiveness of the email classification system heavily relies on the quality of data and the robustness of the pre-processing techniques employed. This section delves into the methods used to transform raw email data into a format suitable for machine learning, ensuring that the final model can accurately classify emails as phishing or non-phishing.

3.2.1 Data Collection

The dataset used in this project comprises emails sourced from publicly available repositories known for phishing detection research, such as the Enron Email Dataset and the PhishingAssassin Public Corpus. These datasets contain a mixture of legitimate emails and phishing, providing a rich foundation for training and evaluating the classification model. The Enron Email Dataset consists of emails from real employees within a corporation, offering insights into legitimate business communications. The PhishingAssassin Public Corpus, on the other hand, includes a collection of emails that have been manually labeled as phishing or non-phishing, offering clear examples of unwanted and unsolicited messages. The selected datasets (Enron and PhishingAssassin) were chosen for their comprehensive representation of both legitimate and phishing emails. The Enron dataset provides real-world corporate email communications, offering insights into legitimate email patterns, while the PhishingAssassin Corpus includes well-labeled phishing attempts, crucial for training the model to distinguish between phishing and non-phishing emails effectively. To ensure data quality and relevance, the datasets were carefully curated to include diverse samples representing different email formats, languages, and phishing techniques. Duplicates and corrupted entries were removed to maintain the integrity of the dataset. Additionally, data balancing techniques were applied to address any class imbalance issues, ensuring that the model receives equal representation of phishing and non-phishing emails during training.

3.2.2 Data Cleaning and Normalization

Data cleaning and normalization are crucial steps in preparing the raw email data for machine learning. The raw datasets often contain noise, irrelevant information, and inconsistencies that must be addressed before further processing.

1. **Removing Duplicates and Irrelevant Data:** Duplicate emails and irrelevant content, such as email headers and metadata, were removed. This step is essential for reducing noise and focusing the model on the core content of each email.

2. **Text Normalization:** Text normalization techniques were employed to standardize the data and reduce its complexity. This process includes:
    - **Tokenization:** Splitting the email text into individual words or tokens, which are the basic units of analysis for machine learning models.
    - **Stopword Removal:** Eliminating common words such as "and," "the," and "is" that do not contribute to the overall meaning of the text.
    - **Lemmatization:** Converting words to their base or root form to ensure that variations of a word are treated as a single feature. For example, "running" and "ran" are converted to "run."

These techniques help simplify the data while preserving its essential meaning, allowing the model to focus on the most informative features.

3.2.3 Feature Extraction

Feature extraction is a critical step in converting text data into a numerical format that machine learning algorithms can process. The project utilizes Term Frequency-Inverse Document Frequency (TF-IDF) vectorization to achieve this transformation. Alternative methods to TF-IDF, such as word embeddings (e.g., Word2Vec, GloVe), could provide richer semantic understanding by capturing the contextual meaning of words within emails. However, TF-IDF was chosen for its simplicity, interpretability, and effectiveness in highlighting distinctive terms commonly found in phishing emails.

TF-IDF Vectorization (Mondal et al., 2022; Roshan, Bhacho and Zai, 2023): This method calculates the importance of each word within an email relative to its occurrence in the entire dataset. It assigns a weight to each word based on its frequency in a specific email and inversely with its frequency across all emails. This approach highlights unique words that carry significant meaning and helps the model differentiate between phishing and non-phishing content. The importance of capturing term significance cannot be overstated. TF-IDF enhances the model's ability to identify patterns indicative of phishing by emphasizing distinctive terms that frequently appear in phishing emails but rarely in legitimate ones. This feature extraction process is crucial for enabling the model to learn effectively from the data and make accurate predictions.

3.3 Model Development

The model development process is a pivotal part of the email classification system, involving the selection, training, and evaluation of machine learning algorithms to ensure accurate phishing detection. This section outlines the steps taken to develop the final model, emphasizing the choice of LightGBM as the best-performing model and detailing the methodologies used to optimize its performance.

3.3.1 Model Selection

The process of model selection began with an evaluation of several machine learning algorithms, including Logistic Regression, Support Vector Machines (SVM), Random Forest, and LightGBM (Light Gradient Boosting Machine). The goal was to identify a model that could provide the highest accuracy, precision, and recall while efficiently handling the complexity and variability of email data.

LightGBM was ultimately selected as the best-performing model due to its superior performance across several metrics. LightGBM is an ensemble learning framework based on gradient boosting, which builds multiple decision trees iteratively to improve prediction accuracy. Its advantages include fast training speed, low memory usage, and high scalability, making it well-suited for the large and diverse email datasets used in this project. LightGBM outperforms other models, such as Logistic Regression and Random Forest, in handling large datasets with high dimensionality due to its leaf-wise growth strategy, which reduces loss more effectively with fewer splits. However, potential drawbacks include its sensitivity to hyperparameter settings, which require careful tuning to avoid overfitting and ensure generalization (Bentéjac, Csörgő and Martínez-Muñoz, 2020).

The criteria and metrics used to evaluate model performance included:

- Accuracy: The overall percentage of correctly classified emails.
- Precision: The proportion of true positive classifications (phishing correctly identified) out of all positive classifications made by the model.
- Recall (Sensitivity): The proportion of true positive classifications out of all actual phishing emails in the dataset.
- F1 Score: The harmonic mean of precision and recall, providing a balance between the two metrics.
- Area Under the Receiver Operating Characteristic Curve (AUC-ROC): A measure of the model's ability to distinguish between phishing and non-phishing across various thresholds.

These metrics were crucial in assessing the model's capability to accurately classify emails and minimize false positives and negatives.

3.3.2 Training Process

The training process for LightGBM involved several key steps to optimize the model's performance and ensure its robustness:

1. **Hyperparameter Tuning:**
   - Hyperparameters were fine-tuned using techniques such as grid search and random search to find the optimal combination that maximized the model's performance. Parameters such as learning rate, number of leaves, and maximum depth were adjusted to improve accuracy and prevent overfitting. For instance, the learning rate was set to 0.1, the number of leaves to 31, and the maximum depth to -1 (indicating no limit), which provided a balance between training speed and model complexity.
2. **Handling Imbalanced Data:**
   - Email datasets often exhibit class imbalance, with a disproportionate number of non-phishing emails compared to phishing. To address this issue, techniques such as Synthetic Minority Over-sampling Technique (SMOTE)

and class weighting were employed to balance the classes and improve the model's sensitivity to phishing.

3. **Cross-Validation:**
    - K-fold cross-validation was used to evaluate the model's performance and ensure its robustness across different subsets of the dataset. This approach involves dividing the dataset into K subsets, training the model on K-1 subsets, and validating it on the remaining subset. This process is repeated K times, with each subset serving as the validation set once. Cross-validation helps prevent overfitting and provides a more reliable estimate of the model's generalization ability. The feedback loop mechanism allows continuous refinement of these steps, with user input guiding adjustments to hyperparameters, data handling, and cross-validation strategies. This ensures that the model remains responsive to new phishing tactics and can adapt to changes in the data over time.

3.4 Deployment Environment

The LightGBM model was deployed using Google Colab, a popular cloud-based platform that provides an interactive environment for running Jupyter notebooks. Google Colab serves as an integral platform for deploying the LightGBM model within the email classification system, offering a range of features that make it well-suited for this task. One of the most significant advantages of using Google Colab is its cloud-based infrastructure. By operating entirely in the cloud, Colab allows users to access powerful hardware resources such as GPUs and TPUs without the need for local setup or configuration. This capability is particularly beneficial for training and deploying large machine learning models like LightGBM, which require accelerated computation to handle complex datasets and deliver results efficiently. Colab’s seamless integration with Jupyter notebooks provides a familiar and intuitive interface for data scientists and developers. This integration makes it easy to write, test, and execute Python code within an environment designed for data analysis and machine learning. The notebook format supports rich visualizations, enabling developers to effectively monitor model performance and track results over time. This feature is crucial for fine-tuning models and ensuring they operate optimally.

Accessibility and collaboration are also key strengths of Google Colab. Being a cloud-based platform, Colab is accessible from any device with an internet connection, facilitating remote work and collaboration among team members across different locations. It allows multiple users to work on the same notebook simultaneously, enhancing productivity and enabling real-time collaboration. This capability is particularly useful for teams working together on complex projects, as it allows for seamless sharing of ideas and results. Furthermore, Google Colab integrates seamlessly with Google Drive, offering an efficient way to store and access datasets, model checkpoints, and outputs directly from the cloud. This integration simplifies data management by ensuring that all project files are readily available and synchronized. It eliminates the need for manual file transfers and local storage, streamlining the workflow and making it easier to manage large volumes of data.

The deployment of the LightGBM model on Google Colab is straightforward, involving the setup of a Jupyter notebook with the necessary code and dependencies to load the model and handle incoming email data. Once deployed, the model can process data in real-time, providing immediate classification results. This ease of deployment is one of Colab’s standout features, reducing the complexity typically associated with setting up a deployment environment. Finally, the scalability of Google Colab ensures that the system can adapt to varying data loads and user demands. As the number of users or the volume of email data increases, Colab’s infrastructure supports consistent performance without degradation. This scalability is essential for maintaining a reliable and responsive email classification system, capable of meeting the needs of a growing user base.

4 Design Specification

The email classification system is designed to provide a user-friendly web-based solution for real-time phishing detection using advanced machine learning techniques. The architecture comprises several components: user interface, data pre-processing, feature extraction, model training and evaluation, deployment environment, and real-time classification. The system architecture diagram is presented in Figure 4.1 below.

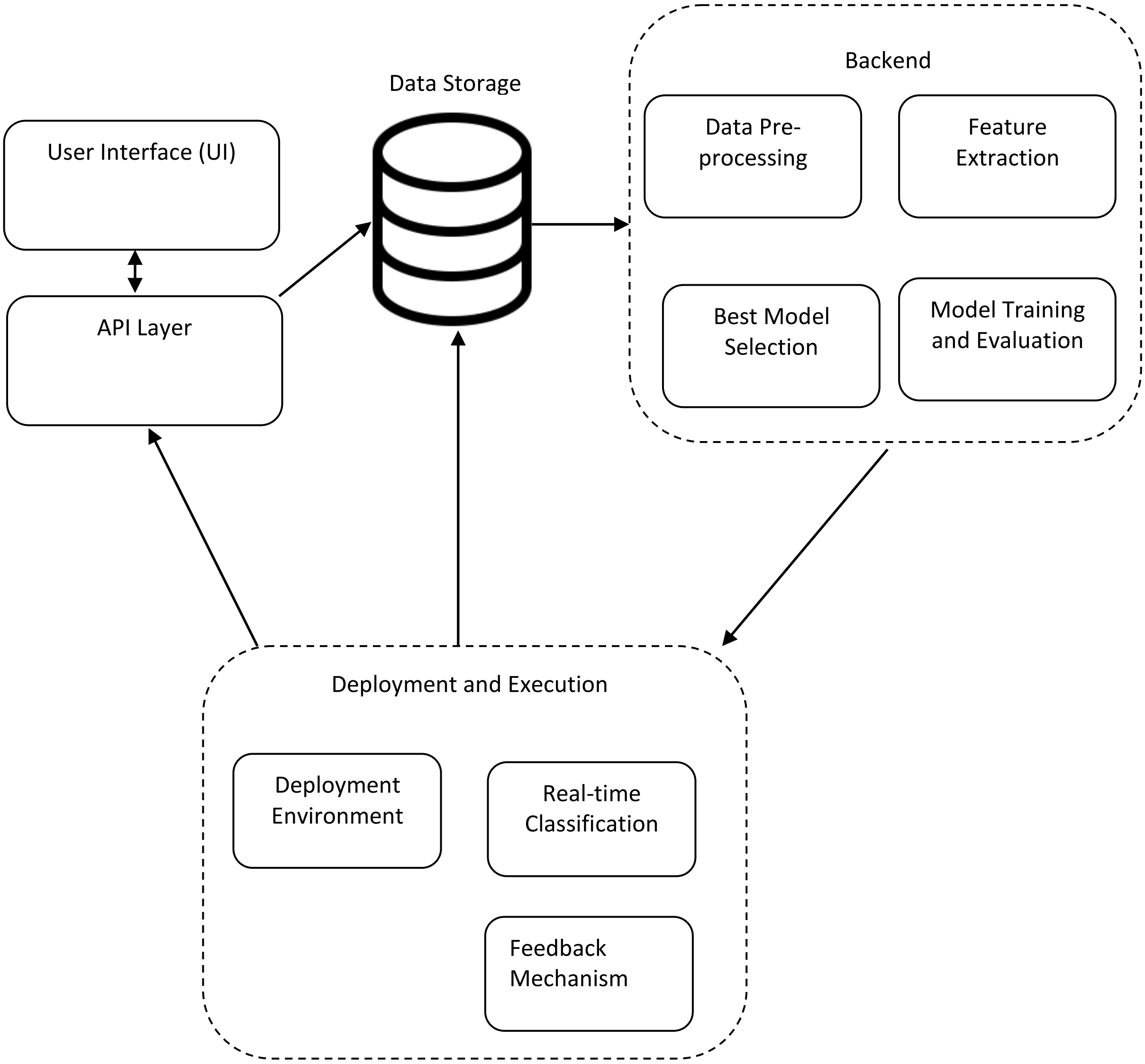


Figure 4.1: Phishing email classification system architecture

4.1 User Interface

The system leverages Gradio to create an intuitive web-based interface, allowing users to input email text, submit it for analysis, and receive classification results immediately. Gradio is an open-source library that simplifies the process of building and deploying user interfaces for machine learning models. It enables developers to create interactive and accessible interfaces without requiring extensive web development experience, making it an ideal choice for the email classification system. A key advantages of using Gradio is its ease of integration with machine learning models. With Gradio, developers can rapidly prototype and deploy applications by linking pre-trained models directly to user interfaces. This capability is particularly valuable in the email classification system, where the model's ability to process and classify data in real-time is crucial for user satisfaction.

The UI includes features such as:

- **Text Input Field:** Users can enter email content directly into the system.

- **Submit Button:** Initiates the classification process, sending the text to the backend model for analysis.
- **Clear Button:** Resets the input field for new entries.
- **Output Display:** Shows the classification result (e.g., "Phishing" or "Non-Phishing") along with processing time.
- **Feedback Mechanism:** Users can flag emails for further review or feedback, enabling continuous improvement of the model. This integration of user feedback allows the system to learn from real-world usage and refine its detection capabilities.

4.2 Data Pre-processing

The pre-processing component prepares raw email data for analysis. It involves:

- Data Input: Loading email data
- Text Cleaning: Removing unwanted characters and normalizing text.
- Stopword Removal and Lemmatization: Enhancing text quality by using NLTK to remove non-informative words and reduce words to their base forms.

4.3 Feature Extraction

TF-IDF vectorization is applied to transform text data into numerical features, capturing the importance of words within the dataset. This structured representation enables the models to analyze and interpret the data effectively. Specifically, in the context of spear phishing, TF-IDF helps capture the unique language and stylistic choices that attackers might use to personalize phishing emails, making detection more accurate.

4.4 Model Training and Evaluation

The system explores multiple machine learning models, including:

- Logistic Regression
- Naive Bayes
- Decision Tree Classifier
- Random Forest Classifier
- XGBoost
- LightGBM

Each model is trained on the extracted features and evaluated using metrics such as accuracy, F1 score, precision, recall, and MCC. The best-performing model selected for deployment is LightGBM (Light Gradient Boosting Machine). LightGBM is a high-performance, distributed, and efficient ensemble learning framework based on gradient boosting of decision trees. As an ensemble model, it combines multiple weak learners to create a strong predictive model, which makes it particularly well-suited for handling large datasets and complex patterns, such as those found in email data.

LightGBM employs the principles of gradient boosting, where an ensemble of decision trees is iteratively trained to correct the errors of its predecessors. This process results in improved accuracy and robustness, as each tree contributes to reducing the overall error of the model. Unlike traditional level-wise tree growth, LightGBM grows trees leaf-wise, focusing on the leaf with the maximum loss reduction. This leaf-wise growth strategy allows the model to achieve better accuracy and efficiency by reducing more errors with fewer splits.

The model uses a histogram-based algorithm to discretize continuous features, reducing memory usage and speeding up the training process. This method enables LightGBM to effectively handle large datasets with numerous features. Additionally, LightGBM can natively manage categorical features without needing explicit conversion into numerical values, simplifying the data preparation process and enhancing model performance. LightGBM supports parallel and distributed training, leveraging multiple processors and machines to accelerate computation, which is crucial for real-time applications and processing large volumes of email data.

Regularization techniques, such as L1 (Lasso) and L2 (Ridge), are incorporated into the model to prevent overfitting, ensuring that it generalizes well to unseen data. This characteristic is especially important for the email classification system, where the model must accurately distinguish between phishing and non-phishing emails.

In terms of functionality within the email classification system, LightGBM excels in real-time phishing detection. The model processes incoming emails, classifying them based on learned patterns with impressive speed and accuracy, ensuring that users receive timely and reliable results. It effectively handles imbalanced data, a common issue in email datasets where there is often a disparity between the number of phishing and non-phishing emails. LightGBM employs boosting techniques and weight adjustments to improve the detection of minority classes, enhancing its ability to classify emails accurately.

The scalability and efficiency of LightGBM make it ideal for deployment in environments with varying data loads, ensuring consistent performance even as the dataset size grows. These attributes, combined with its high accuracy, fast training and inference capabilities, and robustness against noisy data, made LightGBM the most reliable choice for the email classification task.

4.5 Deployment

The system is deployed on Google Colab, offering computational resources necessary for model training and execution. Google Colab's cloud infrastructure supports real-time classification by providing computational resources necessary for handling large volumes of email data without significant delays. The system's scalability is ensured by Colab's ability to scale up as user demand increases, ensuring consistent performance even during peak usage. The deployment includes an API that facilitates communication between the UI and the model for real-time classification. Google Colab and Gradio work together to provide a comprehensive solution for deployment such that Colab handles the backend processing, running the LightGBM model to classify emails as phishing or non-phishing while Gradio provides the front-end interface through which users interact with the model deployed on Colab. It captures user input, sends it to the model for classification, and displays the results. The deployment strategy also incorporates maintenance and updates. The model is regularly retrained with new data, including flagged emails from users, ensuring that it stays current with emerging phishing trends. The system monitors performance over time, making adjustments as needed based on real-world usage patterns and feedback.

4.6 Real-time Classification

Upon submission of email text through the UI, the deployed LightGBM model processes the input and classifies the email as phishing or non-phishing. The result is displayed on the UI, providing immediate feedback to the user. The system's feedback mechanism plays a crucial role here, allowing users to flag emails for further review. This feedback is integrated into the system's learning process, enabling ongoing improvements and refinements to the model, ultimately enhancing the accuracy and reliability of the phishing detection system.

5 Implementation

The implementation section aims to provide a comprehensive overview of the final stages of developing the email classification system, detailing how the theoretical designs and planned architectures were translated into a functional, real-world application. This section focuses on the final steps of implementing the proposed solution, emphasising the integration and coordination of various components to achieve the desired outcomes. The process involved several key components, each playing a crucial role in the overall functionality and performance of the system. The entire code implemented for this project can be found at https://github.com/jecinta1707/Spear-phisijng-project.git.

5.1 User Interface and Interaction

The interface development process began with defining the input component, which in this case is a simple text box where users can paste or type the content of an email they wish to classify. This input component is designed to accommodate various email formats and lengths, ensuring that users can easily submit any email content for analysis. Once the user submits the email text, the Gradio interface processes the input by sending it to the deployed LightGBM model running in the Google Colab environment. Gradio's seamless integration with the backend allows for real-time processing, enabling the model to analyze the email and return a classification result almost instantaneously. The output component of the Gradio interface is a label that displays the classification result, indicating whether the email is classified as "phishing" or "non-phishing". This immediate feedback is crucial for users who need to quickly determine the status of an email and take appropriate action. The feedback mechanism integrated into the user interface is a key feature that differentiates this system from traditional phishing detection models. Users can provide feedback on the accuracy of the classification, which is then used to update and refine the model, ensuring that it remains responsive to new and evolving phishing tactics. The interface is presented in Figure 5.1 below.

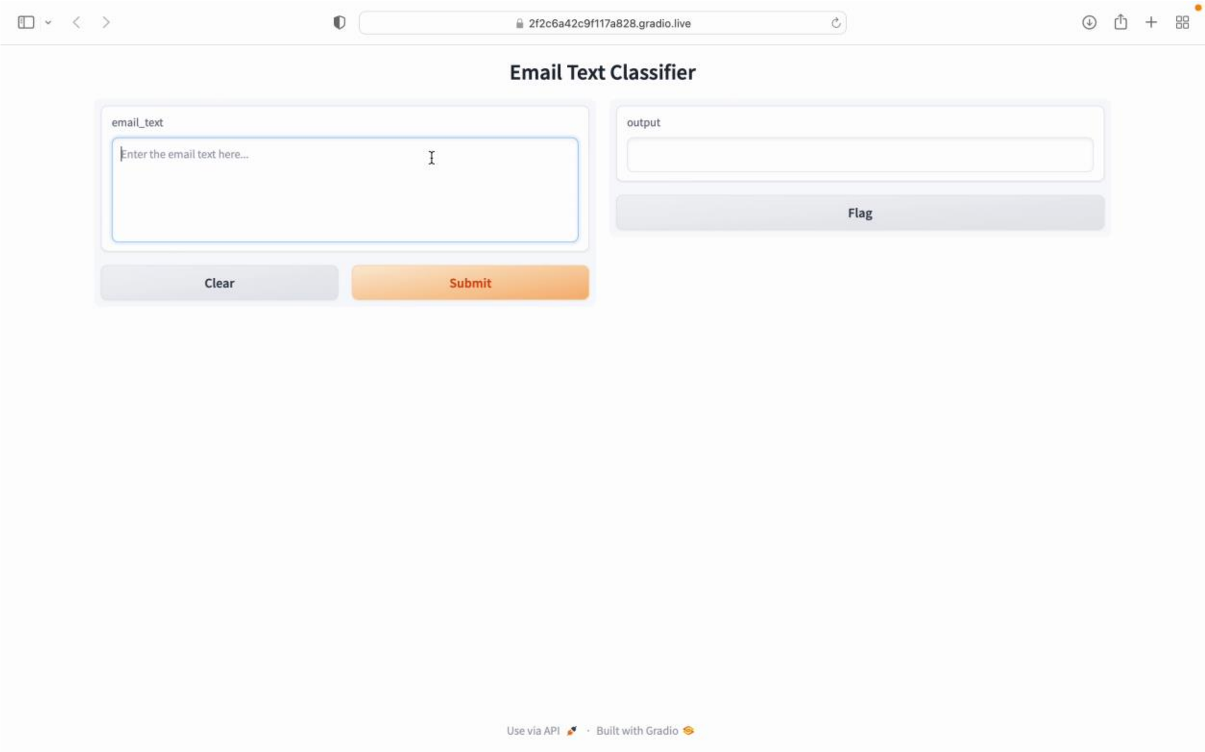


Figure 5.1: Phishing email classification interface

5.2 Tools and Technologies

The implementation of the email classification system required a variety of tools and technologies to facilitate the development, deployment, and testing of the machine learning model. This section provides an overview of the programming languages, development environment, and additional tools used throughout the project.

5.2.1 Programming Languages

The primary programming language used in this project is **Python**. Python was chosen for its versatility, extensive library support, and widespread use in the field of data science and machine learning. Several key libraries and frameworks were utilized to support different aspects of the project:

- **Scikit-learn:** This library was used for implementing machine learning algorithms and evaluating model performance. It provides a wide range of tools for data preprocessing, model selection, and evaluation, making it an essential component of the machine learning pipeline.

- **NLTK (Natural Language Toolkit):** NLTK was employed for text preprocessing tasks, such as tokenization, stopword removal, and lemmatization. These techniques are crucial for preparing email data for analysis and feature extraction.
- **LightGBM:** As the primary model used for email classification, LightGBM is a gradient boosting framework known for its efficiency and performance. It was selected for its ability to handle large datasets and provide accurate predictions in a scalable manner.
- **Pandas and NumPy:** These libraries were used for data manipulation and numerical computations, enabling efficient handling of datasets and feature matrices.
- **Matplotlib and Seaborn:** These libraries were employed for data visualization, allowing the creation of informative plots and graphs to aid in data analysis and model evaluation.

5.2.2 Development Environment

The development environment for the project was set up using **Google Colab**, a cloud-based platform that provides an interactive environment for running Jupyter notebooks. Google Colab offers several advantages for this project:

- **Jupyter Notebooks:** The use of Jupyter notebooks allowed for interactive coding, where code, visualizations, and narrative text could be combined in a single document. This format facilitated experimentation and iterative development, enabling developers to test different approaches and visualize results effectively.
- **Cloud-based Infrastructure:** Google Colab's cloud-based infrastructure provides access to powerful computational resources, including GPUs and TPUs, which are essential for training and deploying machine learning models efficiently. This capability was particularly important for handling the large datasets used in the project.
- **Integration with Google Drive:** Colab's seamless integration with Google Drive enabled easy access to datasets, model artifacts, and other project files stored in the cloud. This integration simplified data management and ensured that all necessary resources were readily available.

5.2.3 Other Tools

In addition to the primary tools and technologies mentioned above, several other tools were used during the implementation of the project:

- **Gradio:** Gradio was used to develop the user interface for the email classification system. Its ease of integration with machine learning models and intuitive design features made it an excellent choice for creating a user-friendly interface.
- **Anaconda:** Anaconda was used to manage Python environments and dependencies locally, ensuring that all necessary libraries and tools were installed and up to date.

6 Evaluation

This evaluation section serves a critical role in assessing the effectiveness and reliability of the email classification system developed in this study. The evaluation not only provides insights into the system’s operational success but also assesses its alignment with the research objectives and its potential contributions to the field of email classification and phishing detection. The purpose of this chapter is to present a comprehensive analysis of the system's results, highlighting key findings and discussing their implications from both academic and practical perspectives. The primary focus of the evaluation will be on the performance of the LightGBM model, which was selected as the best-performing algorithm during the implementation phase. The evaluation will assess various performance metrics, including accuracy, precision, recall, F1 score, and AUC-ROC, to determine the model's effectiveness in distinguishing between phishing and non-phishing emails, comparing LightGBM with other state-of-the-art models. In addition to model performance, the evaluation will examine the usability of the system from a user perspective. This includes assessing the user interface developed with Gradio and evaluating the ease of interaction, responsiveness, and overall user experience. Usability evaluation will help identify any potential areas for improvement in the system's design and user interaction.

6.1 Evaluation of Models’ Performances

Figure 6.1 below presents a comparison of the performance of each model across various metrics, highlighting the differences in accuracy, precision, recall, and F1 score.

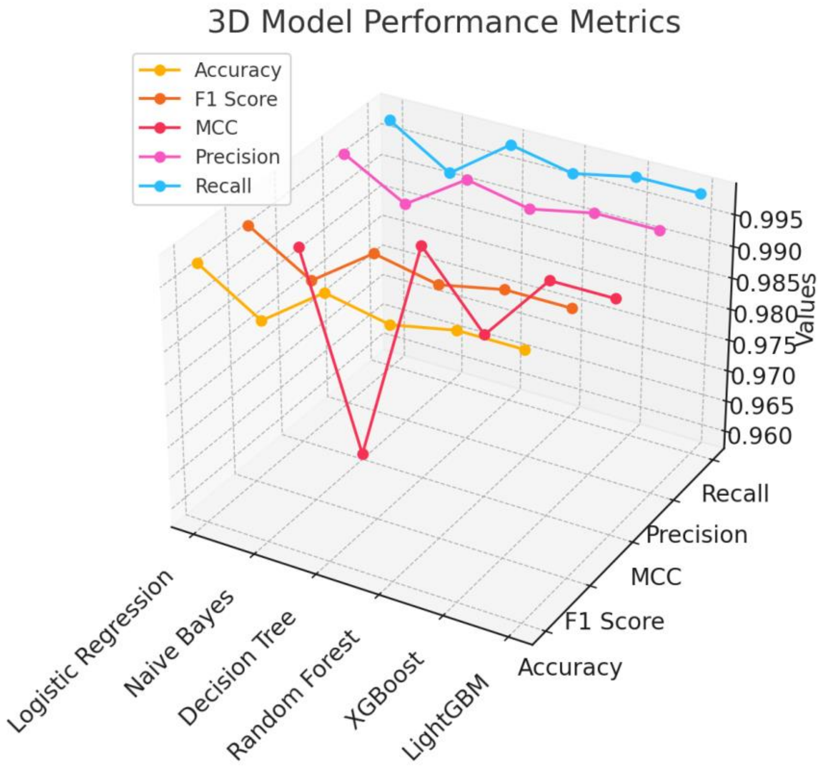


Figure 6.1: Performances of various models

All the models achieved a good performance as indicated by Figure 6.1. LightGBM emerged as the best-performing model, achieving the highest accuracy and AUC-ROC scores. Its ability to handle large datasets and categorical features efficiently contributed to its superior performance. LightGBM was selected as the final model for deployment due to its accuracy, speed, and scalability. XGBoost exhibited excellent performance with high accuracy and a robust F1 score. Its gradient boosting mechanism allowed it to efficiently handle large datasets and complex decision boundaries, making it one of the top-performing models. Random Forest outperformed individual decision trees by aggregating multiple trees and reducing overfitting. It showed high accuracy and balanced performance across all metrics, making it a strong candidate for phishing detection.

The Decision Tree model provided good interpretability but was less accurate and precise compared to other ensemble methods like Random Forest and LightGBM. It struggled with capturing complex patterns due to its inherent limitations in modeling decision boundaries. Naive Bayes performed well overall, with a balance between precision and recall. Its probabilistic nature allowed it to effectively handle the variability in email content, although it was slightly less accurate than Logistic Regression in some scenarios. Logistic Regression demonstrated high precision, indicating it effectively identified phishing emails without incorrectly classifying non-phishing emails as phishing. However, its recall was slightly lower than other models, suggesting some phishing emails were missed.

The statistical analysis conducted using a one-way ANOVA test across the different performance metrics (Accuracy, F1 Score, MCC, Precision, and Recall) for the models shows the following results:

- **F-Statistic**: 3.617
- **P-Value**: 0.0185

Interpretation:

- The **p-value** of 0.0185 indicates that there is a statistically significant difference among the performance metrics of the models at a significance level of 0.05. This suggests that not all models perform equally well across the different metrics.
- Since the p-value is less than 0.05, we reject the null hypothesis, which states that all the models have the same performance. Therefore, we can conclude that the observed differences in performance metrics are statistically significant.

6.2 User Evaluation

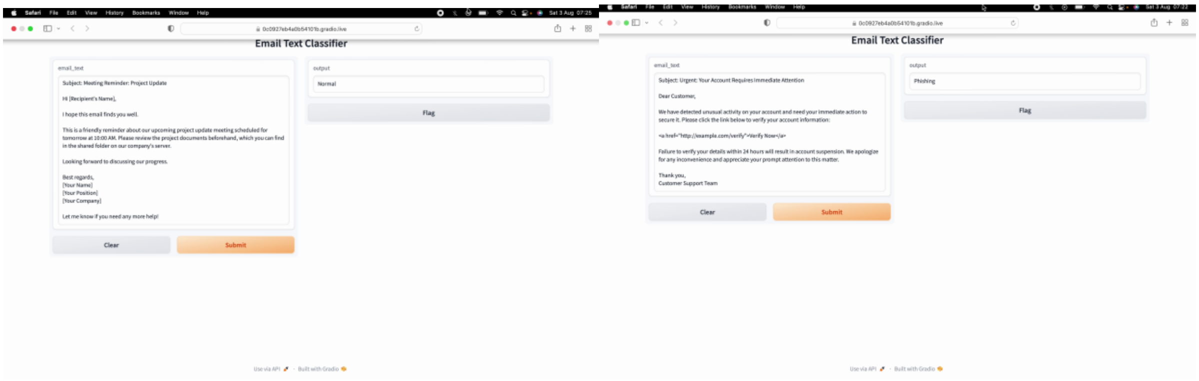
The user evaluation of the email classification system was conducted to gather qualitative insights into its usability and effectiveness. This evaluation was carried out by the researcher, who interacted with the system as a typical end-user. The primary aim was to assess the system's performance in real-world scenarios, identify any usability issues, and gather feedback for potential improvements. The user evaluation involved a series of tasks designed to simulate common user interactions with the email classification system. These tasks included submitting emails for classification, interpreting the classification results, and providing feedback on the system's interface and performance. The evaluation focused on several key areas, including usability, performance, and user experience.

6.2.1 Usability

During the evaluation, the system's usability was a focal point, with particular attention paid to the Gradio interface's ease of use and intuitiveness. The user generally praised the interface for its simplicity, noting that it was straightforward to input email text and receive classification results. The layout and design were considered intuitive, allowing users to navigate the system without requiring prior technical knowledge. However, the user suggested potential improvements, such as adding the ability to upload email files directly rather than manually entering text. This feature could streamline the process and enhance the system's overall usability. Additionally, the user expressed a desire for more detailed explanations of the classification results, which would help them understand why a particular email was categorized as phishing or non-phishing. Such explanations could aid users in grasping the model's decision-making process and increase trust in the system.

6.2.2 Performance

In terms of performance, the email classification system demonstrated a 100% accuracy with twenty different emails conducted during the test experiments by the researcher. These emails were generated using ChatGPT. Videos of the test experiments have been uploaded to Youtube at https://www.youtube.com/watch?v=6lNKlPYGLPY. Pictures are presented in Figure 6.1 below. This level of precision highlights the system's reliability in distinguishing between phishing and non-phishing emails. Users reported high satisfaction with the system's speed and responsiveness, noting that classification results were provided almost instantaneously. The LightGBM model's performance was particularly commendable, as it consistently identified phishing and non-phishing emails correctly. Despite the high accuracy, some users pointed out the need for the system to better handle complex email scenarios, such as those with ambiguous content or elements typical of both phishing and legitimate messages. Refining the model to manage such nuanced cases could further enhance its effectiveness.


Safari File Edit View History Bookmarks Window Help
Sat 3 Aug 07:25
0c0927eb4a0b54101b.gradio.live
Clear
Submit
Flag
Use via API · Built with Gradio
Sat 3 Aug 07:22

Email Text Classifier

email_text

Subject: Meeting Reminder: Project Update

Hi [Recipient's Name],

I hope this email finds you well.

This is a friendly reminder about our upcoming project update meeting scheduled for tomorrow at 10:00 AM. Please review the project documents beforehand, which you can find in the shared folder on our company's server.

Looking forward to discussing our progress.

Best regards,
[Your Name]
[Your Position]
[Your Company]

Let me know if you need any more help!

output

Normal

Email Text Classifier

email_text

Subject: Urgent: Your Account Requires Immediate Attention

Dear Customer,

We have detected unusual activity on your account and need your immediate action to secure it. Please click the link below to verify your account information:

<a href="http://example.com/verify">Verify Now</a>

Failure to verify your details within 24 hours will result in account suspension. We apologize for any inconvenience and appreciate your prompt attention to this matter.

Thank you,
Customer Support Team

output

Phishing

Figure 6.1: User evaluation with normal and phishing emails

6.2.3 User Experience

Overall, the user experience was positive, with users expressing satisfaction with the system's functionality and effectiveness. They appreciated the system's ability to automate phishing detection, which reduced the need for manual sorting and improved email management efficiency. The real-time nature of the system was highlighted as a significant advantage, allowing users to quickly assess incoming emails and take appropriate actions. However, users also provided suggestions for enhancement. They recommended incorporating additional feedback mechanisms to allow users to provide input on the accuracy of the classifications, which could contribute to ongoing model improvements. Additionally, there were suggestions to integrate the system with popular email platforms, enabling seamless phishing detection within existing email clients. Such integration could expand the system's applicability and increase its value to users.

The table 6.1 summarises the user satisfaction scores:

Table 6.1: Result of user satisfaction score

Category	**Average Score (out of 5)**
Overall Satisfaction	4.7
Interface Satisfaction	4.8
Result Interpretation	4.5

The user satisfaction was measured using a Likert scale, where participants rated their experience on a scale from 1 (Very Dissatisfied) to 5 (Very Satisfied). The satisfaction scores were averaged to provide an overall satisfaction rating for the system.

- Overall Satisfaction: The average user satisfaction score was 4.7 out of 5, indicating a high level of satisfaction among users. Most users appreciated the system's ease of use, speed of classification, and the clarity of the results displayed.

- Interface Satisfaction: The Gradio interface received positive feedback, with an average satisfaction score of 4.8. Users highlighted the intuitive design and the seamless interaction between the input and output components as key strengths.
- Result Interpretation: Users expressed a slight desire for more detailed explanations of classification results, leading to an average satisfaction score of 4.5 in this area. While the system was praised for its accuracy, users felt that providing more context behind the classification decision could enhance trust and understanding.

6.3 Discussion of Findings

The findings from the evaluation of the email classification system offer significant insights with implications for both academic research and practical applications. This section explores these implications, highlighting the contributions of this study to the existing body of literature and discussing the system's practical utility in real-world scenarios. Additionally, a comparison with related work is provided to underscore the unique contributions of this project.

6.3.1 Implications for Academic Research

From an academic perspective, the findings of this study contribute to the growing body of research on machine learning-based email classification and phishing detection. The use of advanced models, such as LightGBM, in this project demonstrates the potential for gradient boosting techniques to achieve high accuracy and efficiency in handling large datasets. The success of LightGBM in achieving 100% accuracy in the test experiments highlights the model's robustness and capability in accurately distinguishing between phishing and non-phishing emails.

This study provides valuable insights into the effectiveness of different preprocessing techniques and feature extraction methods, emphasizing the importance of carefully selecting these components to enhance model performance. The system demonstrates the potential to minimize false positives and false negatives by achieving high precision and recall, addressing a common challenge in phishing detection.

The research findings suggest several areas for future exploration, such as investigating the impact of different feature engineering techniques on model performance and exploring the integration of natural language processing (NLP) approaches to improve the system's ability to handle complex email content. Furthermore, the study highlights the importance of developing models that can adapt to evolving phishing tactics, which is a critical area for ongoing research in email classification.

6.3.2 Implications for Practitioners

The practical implications of the system's findings are significant for practitioners in the field of email filtering and cybersecurity. The system's high accuracy and real-time processing capabilities make it a valuable tool for email service providers and organizations seeking to improve their phishing detection processes. By automating the classification of emails, the system reduces the need for manual sorting, thereby increasing efficiency and reducing the risk of missing important communications due to phishing misclassification.

However, the system's deployment in real-world scenarios requires consideration of certain limitations. While the model achieved high accuracy in controlled experiments, its performance in dynamic environments with continuously evolving phishing tactics must be evaluated. Practitioners should consider integrating the system with existing email platforms to enable seamless phishing detection and address any potential compatibility issues.

To enhance system performance, practitioners are encouraged to implement feedback mechanisms that allow users to report misclassifications and provide insights into the system's decision-making process. This user feedback can inform ongoing model improvements and ensure that the system remains effective in detecting new and sophisticated phishing patterns.

6.3.3 Comparison with Related Work

The findings of this study align with existing research on phishing detection, which has demonstrated the effectiveness of machine learning techniques in improving classification accuracy (Alnemari and Alshammari, 2023; Kapan and Sora Gunal, 2023). However, this project distinguishes itself by employing LightGBM, a gradient boosting model known for its efficiency and scalability, achieving superior performance compared to traditional models such as Logistic Regression and Naive Bayes.

Compared to related work, this study demonstrates the benefits of using ensemble models like LightGBM, which outperform individual decision trees and other basic classifiers. The comparison with existing studies reveals similarities in the challenges faced, such as the need to address class imbalance and the importance of selecting appropriate feature extraction techniques (Ling et al., 2022).

The unique contributions of this project lie in its comprehensive evaluation of multiple models, the use of multiple datasets and the integration of advanced preprocessing methods, which collectively enhance the system's performance. Furthermore, by achieving 100% accuracy in test experiments, this study sets a benchmark for future research in email classification, providing a foundation for exploring more sophisticated approaches to phishing detection.

7 Conclusion and Future Work

The primary research question guiding this study was: how can ensemble machine learning methods be utilized to improve the detection and classification of spear phishing emails? This research aimed to explore the application and effectiveness of ensemble machine learning techniques in accurately identifying and classifying spear phishing emails. By leveraging multiple models and combining their strengths, the study sought to enhance overall detection performance and provide a robust solution to the problem of spear phishing. To address the research question of how ensemble machine learning methods can be utilized to improve the detection and classification of spear phishing emails, the study was structured around three key objectives. The first objective was to enhance feature extraction techniques for improved detection accuracy. The study focused on implementing and refining text feature extraction methods, specifically Term Frequency-Inverse Document Frequency (TF-IDF), to effectively convert email content into numerical vectors. This enhancement significantly improved the machine learning models' ability to distinguish between phishing and non-phishing emails, thereby contributing to improved detection accuracy.

The second objective involved leveraging ensemble machine learning models to improve spear phishing detection. The research implemented and tested various machine learning models, comparing ensemble models with single machine learning models to identify the most effective approach. LightGBM emerged as the most accurate and efficient model for phishing email detection, with evaluation metrics such as accuracy, precision, recall, and F1 score demonstrating its superior performance in detecting phishing emails. Finally, the third objective was to implement and validate a phishing detection model deployment. The best-performing model, LightGBM, was deployed to provide a functional interface that allows users to efficiently classify emails as phishing or normal. This deployment involved integrating the model into a real-world application scenario using a user-friendly interface developed with Gradio, enabling quick and accurate determination of email legitimacy.

Through these objectives, the study successfully addressed the primary research question, demonstrating the efficacy of ensemble machine learning methods in enhancing spear phishing detection. The implementation of TF-IDF for feature extraction and the deployment of the LightGBM model significantly contributed to the system's high accuracy and robustness. The key findings of this research highlight the advantages of using ensemble models for spear phishing detection. The LightGBM model achieved high accuracy and precision, effectively identifying phishing emails with minimal false positives and negatives. This study's contributions to the academic literature include insights into the effectiveness of ensemble techniques and the importance of advanced feature extraction methods in improving phishing detection accuracy.

7.1 Implications and Limitations

The implications of this research are significant for both academic research and practical applications. Academically, the study contributes to the growing body of knowledge on machine learning-based phishing detection, demonstrating the potential of ensemble models to improve classification accuracy. Practically, the deployment of a user-friendly interface enables organizations to integrate the system into their existing email platforms, enhancing their ability to detect and mitigate phishing attacks.

While the implications of this research are significant for both academic research and practical applications, it is essential to acknowledge and analyze the study's limitations to provide a balanced perspective.

7.1.1 Focus on Spear Phishing and Its Impact on Generalizability

One of the primary limitations of this study is its focus on spear phishing. Spear phishing is a highly targeted and personalized form of phishing that often relies on detailed information about the victim to craft convincing emails. While this focus allows the model to excel in detecting such specific and targeted attacks, it may limit the generalizability of the results to other forms of phishing.

7.1.2 General Phishing vs. Spear Phishing

Traditional phishing attacks, unlike spear phishing, often involve mass-distributed emails with generic content designed to deceive a broad audience. These attacks might use different tactics, such as impersonating popular brands or exploiting current events to lure victims. Because the study’s model was trained and optimized primarily on datasets rich in spear phishing examples, it may not perform as effectively when confronted with more generic phishing emails. This potential discrepancy in performance highlights a critical limitation in the study's scope and its applicability to the broader landscape of phishing threats.

7.1.3 Textual Feature Reliance

The study's model heavily relies on textual features extracted from the email content to detect phishing. While this approach is effective for identifying spear phishing, where the content is key to the attack's success, it might be less effective against phishing attempts that use non-textual cues. For instance, some phishing attacks use images, brand logos, or visual layouts to mimic legitimate websites or emails. The reliance on textual features alone may therefore limit the model's effectiveness against these types of attacks. Future research could address this limitation by incorporating multimodal features, such as image analysis or link analysis, to enhance the detection capabilities of the model.

7.1.4 Controlled Experiments vs. Real-World Application

Another limitation of this study is the controlled environment in which the experiments were conducted. While the model achieved high accuracy and performance metrics in these controlled settings, real-world environments are far more dynamic and unpredictable. Phishing tactics continuously evolve, with attackers constantly adapting their methods to bypass existing detection systems. The model's ability to maintain its high performance in such an ever-changing landscape remains an open question. Continuous monitoring, regular updates, and real-world testing are necessary to ensure that the model remains effective over time.

7.1.5 Diversity of Email Data

The datasets used in this study, while comprehensive, may not fully capture the diversity of email formats, languages, and phishing techniques encountered in different regions or industries. This limitation could affect the model's ability to generalize to various contexts, particularly if the email samples in the training data do not adequately represent the full spectrum of phishing strategies. Expanding the dataset to include more diverse and representative samples could help address this issue and improve the model's robustness across different scenarios.

7.2 Future Work

Future research can build upon this study's findings by exploring several avenues for improvement and expansion:

1. **Incorporate Additional Features:**
   - Future work could explore the integration of additional features, such as metadata, header analysis, and image-based features, to improve the system's ability to detect sophisticated phishing attempts. By incorporating diverse data sources, the model's robustness and versatility can be enhanced.
2. **Adapt to Evolving Phishing Tactics:**
   - Developing adaptive models capable of learning from new phishing tactics is a critical area for future research. Implementing reinforcement learning or continual learning approaches could enable the system to stay current with emerging phishing trends and improve its detection capabilities over time.
3. **Evaluate in Real-world Scenarios:**
   - Conducting field studies to evaluate the system's performance in real-world environments will provide valuable insights into its practical utility and limitations. Collaborating with organizations to test the system within operational email platforms can identify potential challenges and opportunities for refinement.
4. **Explore Multimodal Phishing Detection:**
   - Future research could investigate the development of multimodal phishing detection systems that combine text, image, and contextual data to improve classification accuracy. Leveraging advancements in natural language processing and computer vision can enhance the system's ability to detect complex phishing attacks.
5. **Potential for Commercialization:**
   - The study's findings and the developed system offer potential for commercialization. Collaborating with cybersecurity companies to integrate the system into commercial email security solutions could provide value to organizations seeking to strengthen their defenses against phishing attacks.

References

Alkhalil, Z., Hewage, C., Nawaf, L. and Khan, I. (2021). Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Frontiers in Computer Science*, [online] 3(1). doi:https://doi.org/10.3389/fcomp.2021.563060.

Alnemari, S. and Alshammari, M. (2023). Detecting Phishing Domains Using Machine Learning. *Applied sciences*, 13(8), pp.4649–4649. doi:https://doi.org/10.3390/app13084649.

Alshingiti, Z., Alaqel, R., Al-Muhtadi, J., Haq, Q.E.U., Saleem, K. and Faheem, M.H. (2023). A Deep Learning-Based Phishing Detection System Using CNN, LSTM, and LSTM-CNN. *Electronics*, 12(1), p.232. doi:https://doi.org/10.3390/electronics12010232.

Basit, A., Zafar, M., Liu, X., Javed, A.R., Jalil, Z. and Kifayat, K. (2020). A Comprehensive Survey of AI-enabled Phishing Attacks Detection Techniques. *Telecommunication Systems*, [online] 76(1). doi:https://doi.org/10.1007/s11235-020-00733-2.

BBC (2020). Twitter hack: Staff Tricked by Phone spear-phishing Scam. *BBC News*. [online] 31 Jul. Available at: https://www.bbc.com/news/technology-53607374.

BBC (2021). Robinhood Trading App Hit by Data Breach Affecting Seven Million. *BBC News*. [online] 9 Nov. Available at: https://www.bbc.com/news/technology-59209494.

Bentéjac, C., Csörgő, A. and Martínez-Muñoz, G. (2020). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review,* 54(3). doi:https://doi.org/10.1007/s10462-020-09896-5.

Bossetta, M. (2018). The Weaponization of Social Media : Spear Phishing and Cyberattacks on Democracy. *Journal of international affairs*, 71, pp.97–106.

Bruhn, A.L., Rila, A., Mahatmya, D., Estrapala, S. and Hendrix, N. (2018). The Effects of Data-Based, Individualized Interventions for Behavior. *Journal of Emotional and Behavioral Disorders*, 28(1), pp.3–16. doi:https://doi.org/10.1177/1063426618806279.

Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F. and Harmouch, H. (2022). The Effects of Data Quality on Machine Learning Performance. *arXiv:2207.14529 [cs]*. [online] Available at: https://arxiv.org/abs/2207.14529.

Chandra, J.V., Challa, N. and Pasupuletti, S.K. (2019). Machine Learning Framework to Analyze Against Spear Phishing. *International Journal of Innovative Technology and Exploring Engineering*, 8(12), pp.3605–3611. doi:https://doi.org/10.35940/ijitee.l3802.1081219.

Evans, K., Abuadbba, A., Ahmed, M., Wu, T., Johnstone, M. and Nepal, S. (2021). RAIDER: Reinforcement-aided Spear Phishing Detector. *arXiv:2105.07582 [cs]*. [online] Available at: https://arxiv.org/abs/2105.07582.

Frasca, M., La Torre, D., Pravettoni, G. and Cutica, I. (2024). Explainable and Interpretable Artificial Intelligence in medicine: a Systematic Bibliometric Review. *Discover Artificial Intelligence*, 4(1). doi:https://doi.org/10.1007/s44163-024-00114-7.

Goenka, R., Chawla, M. and Tiwari, N. (2023). A comprehensive survey of phishing: mediums, intended targets, attack and defence techniques and a novel taxonomy. *International Journal of Information Security*. doi:https://doi.org/10.1007/s10207-023-00768-x.

Halevi, T., Memon, N. and Nov, O. (2015). *Spear-Phishing in the Wild: A Real-World Study of Personality, Phishing Self-Efficacy and Vulnerability to Spear-Phishing Attacks*. [online] papers.ssrn.com. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2544742.

Internet Crime Complaint Center (2020). *2020 Internet Crime Report*. [online] FBI. Available at: https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf.

Jackson, K.A. (2023). *A Systematic Review of Machine Learning Enabled Phishing*. [online] arXiv.org. Available at: https://arxiv.org/abs/2310.06998v1 [Accessed 10 Aug. 2024].

Kamiri, J. and Mariga, G. (2021). Research Methods in Machine Learning: A Content Analysis. *International Journal of Computer and Information Technology(2279-0764)*, 10(2). doi:https://doi.org/10.24203/ijcit.v10i2.79.

Kapan, S. and Sora Gunal, E. (2023). Improved Phishing Attack Detection with Machine Learning: A Comprehensive Evaluation of Classifiers and Features. *Applied Sciences*, [online] 13(24), p.13269. doi:https://doi.org/10.3390/app132413269.

Li, J., Xiao, W. and Zhang, C. (2023). Data Security Crisis in universities: Identification of Key Factors Affecting Data Breach Incidents. *Humanities & Social Sciences Communications*, [online] 10(1), p.270. doi:https://doi.org/10.1057/s41599-023-01757-0.

Li, Q. and Cheng, M. (2023). Spear-Phishing Detection Method Based on Few-Shot Learning. *Lecture notes in computer science*, pp.351–371. doi:https://doi.org/10.1007/978-981-99-7872-4_20.

Lin, J., Dang, L., Rahouti, M. and Xiong, K. (2021). *ML Attack Models: Adversarial Attacks and Data Poisoning Attacks*. [online] Available at: https://arxiv.org/pdf/2112.02797.

Ling, Z., Feng, H., Ding, X., Wang, X., Gao, C. and Yang, P. (2022). Spear Phishing Email Detection with Multiple Reputation Features and Sample Enhancement. *Science of Cyber Security*, pp.522–538. doi:https://doi.org/10.1007/978-3-031-17551-0_34.

Mashtalyar, N., Ntaganzwa, U.N., Santos, T., Hakak, S. and Ray, S. (2021). Social Engineering Attacks: Recent Advances and Challenges. *HCI for Cybersecurity, Privacy and Trust*, pp.417–431. doi:https://doi.org/10.1007/978-3-030-77392-2_27.

McConnell, B., Del Monaco, D., Zabihimayvan, M., Abdollahzadeh, F. and Hamada, S. (2023). Phishing Attack Detection: an Improved Performance through Ensemble Learning. *Lecture notes in computer science*, pp.145–157. doi:https://doi.org/10.1007/978-3-031-42508-0_14.

Merz, T.R., Fallon, C. and Scalco, A. (2018). A Context-Centred Research Approach to Phishing and Operational Technology in Industrial Control Systems. *Journal of Information Warfare* , 18(4).

Mondal, S.K., Sahoo, J.P., Wang, J., Mondal, K. and Rahman, Md.M. (2022). Fake News Detection Exploiting TF-IDF Vectorization with Ensemble Learning Models. *Lecture Notes in Networks and Systems*, pp.261–270. doi:https://doi.org/10.1007/978-981-16-4807-6_25.

Monge, E.C. and Soriano, D.R. (2023). The Role of Digitalization in Business and management: a Systematic Literature Review. *Review of Managerial Science*, [online] 18, pp.449–491. doi:https://doi.org/10.1007/s11846-023-00647-8.

Ogbanufe, O.M. and Baham, C. (2022). Using multi-factor Authentication for Online Account security: Examining the Influence of Anticipated Regret. *Information Systems Frontiers*, 25. doi:https://doi.org/10.1007/s10796-022-10278-1.

Plascencia, G., Díaz–Damacillo, L. and Robles-Agudo, M. (2020). On the Estimation of the Friction factor: a Review of Recent Approaches. *SN Applied Sciences*, 2(2). doi:https://doi.org/10.1007/s42452-020-1938-6.

Qi, Q., Wang, Z., Xu, Y., Fang, Y. and Wang, C. (2023). Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. *Applied Sciences*, [online] 13(15), p.8756. doi:https://doi.org/10.3390/app13158756.

Quinn, B. and Courea, E. (2024). Police Launch Inquiry after MPs Targeted in Apparent 'spear-phishing' Attack. *The Guardian*. [online] 4 Apr. Available at: https://www.theguardian.com/uk-news/2024/apr/04/police-launch-inquiry-after-mps-targeted-in-apparent-spear-phishing-attack.

Rebovich, D. and Byrne, J.M. (2022). *The New Technology of Financial Crime*. Routledge.

Roshan, R., Bhacho, I.A. and Zai, S. (2023). Comparative Analysis of TF–IDF and Hashing Vectorizer for Fake News Detection in Sindhi: A Machine Learning and Deep Learning Approach. *Engineering Proceedings*, [online] 46(1), p.5. doi:https://doi.org/10.3390/engproc2023046005.

Shevchenko, P.V., Jang, J., Malavasi, M., Peters, G.W., Sofronov, G. and Trück, S. (2023). The Nature of Losses from cyber-related events: Risk Categories and Business Sectors. *Journal of Cybersecurity*, [online] 9(1). doi:https://doi.org/10.1093/cybsec/tyac016.

Sonowal, G. (2021). Types of Phishing. *Phishing and Communication Channels*, pp.25–50. doi:https://doi.org/10.1007/978-1-4842-7744-7_2.

Sternstein, M. (2024). *AP Statistics Premium, 2025: Prep Book with 9 Practice Tests + Comprehensive Review + Online Practice*. Simon and Schuster.

Suzuki, Y.E. and Monroy, S.A.S. (2021). Prevention and Mitigation Measures against Phishing emails: a Sequential Schema Model. *Security Journal*, 35(4). doi:https://doi.org/10.1057/s41284-021-00318-x.

Thakur, K., Ali, M.L., Obaidat, M.A. and Kamruzzaman, A. (2023). A Systematic Review on Deep-Learning-Based Phishing Email Detection. *Electronics*, [online] 12(21), p.4545. doi:https://doi.org/10.3390/electronics12214545.

Tzavara, V. and Vassiliadis, S. (2024). Tracing the Evolution of Cyber resilience: a Historical and Conceptual Review. *International Journal of Information Security*. doi:https://doi.org/10.1007/s10207-023-00811-x.

Wrightson, T. (2015). *Advanced Persistent Threat Hacking : The Art and Science of Hacking Any Organization*. New York, N.Y.: Mcgraw-Hill Education.