# Combating Phishing and Social Engineering Using Machine Learning and NLP

Enhancing User Protection Against Email-Based Cyber Threats

Koray Aman Arabzadeh

**MID SWEDEN UNIVERSITY**
**Department of Information Systems and Technology**

**Examiner:** Mikael Hasselmalm, mikael.hasselmalm@miun.se
**Supervisor:** Mikael Hasselmalm, mikael.hasselmalm@miun.se
**Author:** Koray Aman Arabzadeh, amar2100@student.miun.se
**Study Program:** TDATG, Bachelor of Science with a major in Computer Engineering, 180 ECTS
**Main Field of Study:** Bachelor of Science with a major in Computer Engineering.
**Spring,** 2024

# Abstract

As the digital era progresses, cybersecurity threats become increasingly sophisticated, particularly those involving phishing and social engineering via email. These threats exploit both technical and human vulnerabilities, necessitating new, innovative defense methods.

This project introduces a groundbreaking approach using machine learning and natural language processing (NLP) to effectively identify and neutralize phishing attempts and deceptive messages. By applying algorithms such as Random Forest Classifier and Gradient Boosting Classifier, the ability to detect complex threat patterns is significantly improved. Logistic regression is employed to model probabilities, facilitating rapid and efficient identification of potential cyber threats based on email content and distinct features.

Additionally, the project delivers a web application developed in Flask, providing an intuitive interface for real-time testing of the model's efficiency in recognizing phishing messages. This research highlights the extensive capacity of machine learning and NLP techniques to counter the continuously evolving wave of cyber threats, proving to be an effective strategy for enhancing digital security and protecting users from sophisticated attack methods.

**Keywords:** digital transformation, cybersecurity landscape, email phishing, social manipulation, machine learning techniques, natural language processing (NLP), Random Forest Classifier, Gradient Boosting Classifier, binary classification, Tf-idf, real-time phishing detection, advanced persistent threats (APT), imbalanced datasets, and pattern recognition.

# Sammanfattning

I takt med att den digitala eran fortskrider, blir cybersäkerhetshoten allt mer sofistikerade, särskilt de som involverar phishing och social engineering via e-post. Dessa hot, som utnyttjar både tekniska och mänskliga sårbarheter, kräver nya, innovativa försvarsmetoder.

Detta projekt presenterar en banbrytande metod som använder sig av maskininlärning och NLP för att effektivt identifiera och neutralisera försök till phishing och bedrägliga meddelanden. Genom att tillämpa algoritmer som RandomForestClassifier och GradientBoostingClassifier, förbättras förmågan att identifiera komplexa hotmönster avsevärt. Logistisk regression används för att modellera sannolikheter och underlätta snabb och effektiv identifiering av potentiella cyberhot baserat på e-postinnehåll och distinkta egenskaper.

Projektet bidrar också med en webbapplikation utvecklad i Flask, som erbjuder ett intuitivt gränssnitt för realtidstestning av modellens effektivitet i att känna igen phishingmeddelanden. Denna forskning belyser den omfattande kapaciteten hos maskininlärning och NLP-tekniker för att motverka den ständigt utvecklande vågen av cyberhot, vilket visar sig vara en effektiv strategi för att stärka digital säkerhet och skydda användare från avancerade angreppsmetoder.

**Nyckelord:** digital transformation, cybersäkerhetslandskap, e-postphishing, social manipulation, maskininlärningstekniker, naturlig språkbehandling (NLP), RandomForestClassifier, GradientBoostingClassifier, binär klassificering, Tf-idf, realtidsdetektion av phishing, avancerade persistenta hot (APT), obalanserade dataset, och mönsterigenkänning.

# Preface

This project represents a significant step in my understanding of cybersecurity challenges, with a specific focus on phishing and social engineering, motivated by the increased risks internet users face daily.

A heartfelt thank you is extended to my teacher, Mikael Hasselmalm, for his guidance and support, as well as to my friends Jesper Nilsson and Lucas Persson for their valuable feedback and exchange of ideas.

My aim is for this work to enrich the academic community and inspire new solutions for a safer digital future.

I look forward to applying the insights gained in my continued career.

Sundsvall, 03/2024 Koray Aman Arabzadeh

# Table of Contents

# Terminology

Cyber Threats: Various harmful activities within digital environments.

NLP (Natural Language Processing): Methods enabling machines to interpret human language.

AI (Artificial Intelligence): Systems capable of tasks requiring human-like intelligence.

ChatGPT: An AI tool designed for creating text similar to human dialogue.

ML (Machine Learning): The process where computers improve through data.

Phishing: A digital hazard involving deceitful impersonation.

Social Engineering: The art of exploiting human psychological vulnerabilities.

TF-IDF (Term Frequency-Inverse Document Frequency): A metric assessing a term's significance in documents.

Logistic Regression: A predictive analysis for determining the probability of occurrences.

RandomForestClassifier: A method utilizing multiple decision trees to enhance prediction precision.

GradientBoostingClassifier: A strategy to refine models by prioritizing correction of mistakes.

# 1 Introduction

In the ever-evolving digital landscape, the sophistication of cybersecurity threats, particularly those involving phishing and social engineering via email, presents a formidable challenge. These insidious threats exploit not only technological vulnerabilities but also human psychological weaknesses, making detection and counteraction increasingly difficult. The amalgamation of technology and psychology in these attacks necessitates innovative solutions that surpass traditional security measures.

This project addresses the critical need for advanced defense mechanisms by introducing a novel approach that utilizes the capabilities of Machine Learning (ML) and Natural Language Processing (NLP) to enhance email security. The methodology employs cutting-edge algorithms, including RandomForestClassifier and GradientBoostingClassifier, alongside logistic regression, to accurately identify and neutralize phishing attempts and deceptive messages. By analyzing linguistic patterns and distinctive features of email communications, the solution aims to predict and mitigate potential cyber threats with unprecedented efficiency.

The significance of this research lies in its potential to drastically reduce the success rate of phishing and social engineering attacks, thereby safeguarding individuals and organizations from their damaging consequences. Furthermore, the development of a Flask-based web application provides an accessible platform for real-time phishing detection, embodying the practical application of research findings. This project not only contributes to the academic discourse on cybersecurity but also offers a scalable solution that can be seamlessly incorporated into existing digital security frameworks, enhancing protection against sophisticated cyber threats.

In crafting this innovative approach, a critical gap in the cybersecurity domain is addressed, offering a proactive tool against the constantly evolving tactics of cybercriminals. The implications of this work are far-reaching, promising a future where digital communications are more secure, and users are better equipped to navigate the complexities of the online world with confidence.

# 2 Theory

This section explores phishing's multifaceted nature within the digital domain and the technological innovations aimed at its mitigation. Through a deep dive into cyber threats and the application of cutting-edge tools like NLP and ML, it unveils strategies to bolster cybersecurity. The journey encapsulates both the challenges and solutions, equipping readers with insights into combating digital deception effectively.

To gather inspiration and guidance, the project has benefited from online resources. A significant contribution has been the article "Phishing Detection met Generative AI" published on Medium [1], which offers insights into the use of generative AI to detect phishing attacks. This article has been a valuable source of inspiration and has contributed to the project's methodological approach. Another important reference is the scientific article "A Phishing-Attack-Detection Model Using Natural Language Processing and Deep Learning" [2] published on MDPI. This research highlights the potential of NLP and deep learning to improve phishing attack detection, which is directly relevant to the project.

## 2.1 Cyber Threats

Cyber threats pose risks from digital media, such as the internet and networks, aimed at individuals and organizations to steal or damage data. The 2020 Cybersecurity Report in Sweden identifies state actors and criminal groups as the main sources of these threats. State actors aim to carry out sophisticated cyberattacks for political and economic objectives, while criminals seek financial gain through phishing. The report emphasizes the need for strong security measures and trained cybersecurity personnel to counter these threats, necessitating national and international efforts. [3]

## 2.2 Social Engineering

Social engineering is a complex method of psychological manipulation. that exploits human weaknesses rather than technical flaws to access information, systems, or premises. It can take many forms, including phishing, where for example attackers send fake emails pretending to be from legitimate sources to trick recipients into revealing personal information, and pretexting, where attackers create a fabricated scenario to deceive victims into disclosing information. An effective defense strategy against social engineering requires both technical measures, such as security software and firewalls, and strong awareness among employees. Education and consistent reminders about the dangers of social engineering and how to identify manipulation attempts are essential for safeguarding individuals and organizations from such attacks. [4]

## 2.3 Natural Language Processing

NLP is the cornerstone of this project, serving as a bridge between human language and computers' ability to process it. The project focuses on identifying and classifying phishing emails by analyzing and interpreting the content of email messages. NLP techniques such as tokenization, lemmatization, and sentiment analysis are employed to extract and interpret meaningful information from texts, allowing for the accurate identification of potentially harmful emails with higher precision. The challenge for NLP is dealing with the ambiguity and subtle nuances of language, requiring sophisticated algorithms and models. The project aims to develop and implement robust NLP models that can effectively differentiate between genuine and fraudulent communications, essential for enhancing cybersecurity and protecting users against increasingly sophisticated phishing attacks. [5]

## 2.4 Machine Learning

ML is utilized in this project to detect and classify phishing emails. The ML model is trained on labeled datasets, learning to associate text characteristics with their labels and developing the ability to predict the label for new, unseen texts. By analyzing data sets, ML algorithms learn to recognize patterns and characteristics common to phishing messages, enabling the automatic identification of potential threats. The ability to learn from past data and apply this knowledge to new, unknown data sets makes ML particularly powerful for cybersecurity applications. Specific ML techniques, such as logistic regression, random forest, and gradient boosting, are applied to refine detection precision and efficiency. [6]

## 2.5 Logistic Regression

Logistic regression is another statistical method used to model the probability of a certain outcome variable, typically in binary terms such as yes/no, 0/1, or true/false. In the context of phishing webpage detection, logistic regression is used to evaluate and classify web pages as legitimate or phishing by analyzing various characteristic features. This method helps improve the accuracy of identifying phishing pages by providing a quantitative assessment of the likelihood that a given webpage or email is malicious. [7]

## 2.6 Random Forest Classifier

Random Forest is also used in this project to distinguish between legitimate and malicious emails. RF, a popular machine learning algorithm developed by Leo Breiman and Adele Cutler, combines several decision trees to provide more reliable results in classification and regression problems. By randomly selecting features and data, Random Forest reduces the risk of overfitting and makes predictions more precise. [8] This method is especially useful in areas requiring high precision and reliability in the classification process, such as phishing site detection. [9]

## 2.7 Gradient Boosting Classifier

Gradient Boosting Classifier (GBC) represents a potent method within the machine learning domain, utilized in this project to efficiently identify and differentiate between legitimate and phishing-related emails. This technique is based on the idea of progressively enhancing a model's predictive capability by iteratively adding weak classifiers. Each new classifier focuses on correcting the mistakes made by its predecessors, resulting in gradual improvement and a strong composite model. What makes GBC particularly suitable for the task of phishing detection is its ability to discern subtle patterns and signals that differentiate malicious data from harmless, even when the differences are minimal. Through this method, a system that is both sensitive and precise can be developed, minimizing false positives while effectively capturing real threats. [10]

## 2.8 Term Frequency-Inverse Document Frequency

TF-IDF is used in this project for natural language processing, playing a crucial role in analyzing and understanding text data content. This technique combines two key metrics – the frequency of a word's occurrence in a specific text (Term Frequency) and the uniqueness of the word across multiple texts (Inverse Document Frequency) – to assign a weighted importance to each word in the text. This method allows for quantifying the significance of words in relation to a larger document set, making it possible to distinguish which terms are most characteristic of a specific document. In the context of phishing detection, TF-IDF is used to efficiently identify and flag words or phrases that commonly occur in phishing messages but are rare in legitimate emails, facilitating more precise classifications of potentially harmful content. [11]

## 2.9 Python

For this initiative, Python has been chosen, a programming language renowned for its wide range of libraries, due to its strengths in the areas of machine learning and natural language processing. Its broad array of libraries and frameworks, including NumPy, SpaCy, Pandas, Scikit-learn, and NLTK, offer powerful resources for streamlined data handling, thorough analysis, and the creation of sophisticated models. Python significantly simplifies the process of implementing and fine-tuning various algorithms, establishing the language as an optimal tool for tackling challenges within cybersecurity, especially in the fight against phishing. [12]

## 2.10 Balanced Data

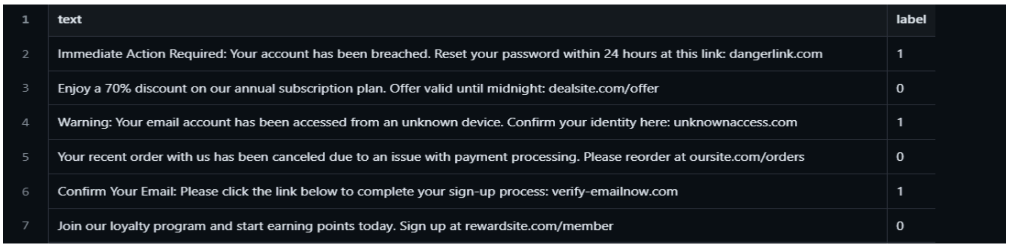
In machine learning, and specifically for classification problems such as phishing identification, the presence of datasets where classes are not evenly distributed can result in biased models that tend to favor the predominant class. Encord highlights the importance of balancing datasets to prevent such biases, which is crucial for the model to function effectively in real-world applications. Balanced datasets ensure that all classes are adequately represented, allowing the model to learn the distinctive characteristics of each class without being overwhelmed by the majority class. This enhances the model's ability to generalize and perform accurately on previously unseen data. [13]

# 3 Method

This project deploys NLP and ML techniques within a Flask-based web application to distinguish between legitimate and phishing emails. The method section outlines a comprehensive approach, from data collection and doing the preprocessing to model training and validation. By meticulously curating a balanced dataset and selecting advanced algorithms, the project aims to refine phishing detection capabilities. The methodology is designed to ensure accuracy, efficiency, and user-friendliness, encapsulating the technical journey from raw data to a reliable classification tool. Through this endeavor, the project enhances digital security by providing an effective solution against phishing threats.

## 3.1 Data Collection and Processing

To support model training, an extensive dataset was compiled using ChatGPT 3.5 and Gemini Google AI, containing email texts labeled as "legitimate" (0) or "phishing" (1). The dataset reflects a varied but balanced mix of email messages, contributing to well-founded model training. See figure 1.



| text | label |
|---|---|
| Immediate Action Required: Your account has been breached. Reset your password within 24 hours at this link: dangerlink.com | 1 |
| Enjoy a 70% discount on our annual subscription plan. Offer valid until midnight: dealsite.com/offer | 0 |
| Warning: Your email account has been accessed from an unknown device. Confirm your identity here: unknownaccess.com | 1 |
| Your recent order with us has been canceled due to an issue with payment processing. Please reorder at oursite.com/orders | 0 |
| Confirm Your Email: Please click the link below to complete your sign-up process: verify-emailnow.com | 1 |
| Join our loyalty program and start earning points today. Sign up at rewardsite.com/member | 0 |

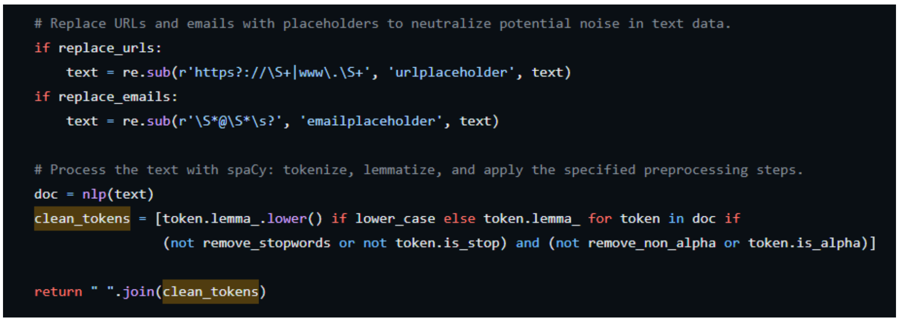
*Figure 1: Displays a CSV file with data generated using ChatGPT 3.5 and Gemini Google AI.*

### 3.1.1 Data Collection

The data has been compiled to ensure a balanced representation of both legitimate messages and phishing attempts. This diversity in data is crucial for an unbiased and efficient learning process.

### 3.1.2 Data Preprocessing

The preparation of text data involves several steps, including text cleaning, tokenization, and normalization, removal of stopwords, and lemmatization. These steps ensure that the data is optimized for processing by ML models. The code is shown in figure 2.



```
# Replace URLs and emails with placeholders to neutralize potential noise in text data.
if replace_urls:
    text = re.sub(r'https?://\S+|www\.\S+', 'urlplaceholder', text)
if replace_emails:
    text = re.sub(r'\S*@\S*\s?', 'emailplaceholder', text)

# Process the text with spaCy: tokenize, lemmatize, and apply the specified preprocessing steps.
doc = nlp(text)
clean_tokens = [token.lemma_.lower() if lower_case else token.lemma_ for token in doc if
                (not remove_stopwords or not token.is_stop) and (not remove_non_alpha or token.is_alpha)]

return " ".join(clean_tokens)
```

*Figure 2 illustrates the process of tokenization and normalization, removal of stopwords, and lemmatization.*

**Text Cleaning:** Involves eliminating irrelevant text and symbols, and standardizing URLs to minimize noise.

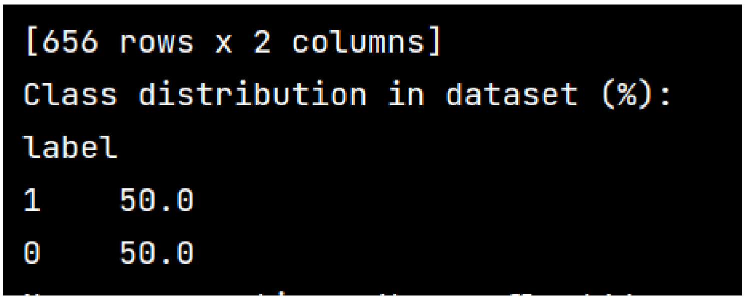
**Tokenization and Normalization:** Involves breaking down text into individual words and converting them to lowercase to create a uniform dataset.

**Stopwords Removal:** Involves removing common words that do not contribute to the meaning.

**Lemmatization:** Entails transforming words into their fundamental form to simplify and strengthen the model's ability to generalize.

### 3.1.3 Generation of Balanced Data

To avoid the common challenges associated with unbalanced datasets, which can lead to model bias, a method was chosen where balanced data, which is shown in the figure 3, was generated using ChatGPT and Gemini. By carefully curating the dataset to ensure an even distribution of the classes "legitimate" (0) and "phishing" (1), the model's learning process was optimized to handle both categories with equivalent precision, as shown in figure 3. This strategy not only contributed to increasing the model's accuracy but also its ability to generalize over unseen data, as evidenced by the improved performance metrics. Opting for a balanced dataset from the start proved crucial for the project's goal of creating a robust model for classifying email messages

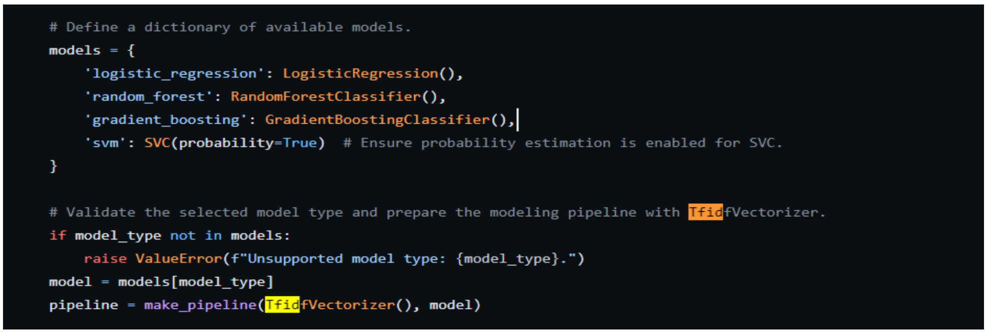


```
[656 rows x 2 columns]
Class distribution in dataset (%):
label
1     50.0
0     50.0
```

*Figure 3: Demonstrates that exactly 50% for both labels "1" (phishing) and "0" (legitimate), indicating the data is balanced.*

## 3.2 Model Training and Validation

After processing with the TF-IDF technique as shown in Figure 4, the text data is prepared for analysis using three carefully selected machine learning models: Logistic Regression, Random Forest, and Gradient Boosting. These models, already trained on a balanced dataset of labeled email messages, are used to evaluate new texts and predict the probability that an email is phishing.



```python
# Define a dictionary of available models.
models = {
    'logistic_regression': LogisticRegression(),
    'random_forest': RandomForestClassifier(),
    'gradient_boosting': GradientBoostingClassifier(),
    'svm': SVC(probability=True)  # Ensure probability estimation is enabled for SVC.
}

# Validate the selected model type and prepare the modeling pipeline with TfidfVectorizer.
if model_type not in models:
    raise ValueError(f"Unsupported model type: {model_type}.")
model = models[model_type]
pipeline = make_pipeline(TfidfVectorizer(), model)
```

*Figure 4: Processing with the TF-IDF technique prepares the text data.*

### 3.2.1 Visualization and Comparison:

To illustrate the models' predictive capabilities, a script was implemented to visualize the probability that a set of email examples is phishing. This allows not only to see each model's assessment in numerical form but also to visually compare these assessments through bar plots. Each plot presents the model's assessment of the probability of phishing for both legitimate and suspicious email messages, providing insight into their discriminatory power.

### 3.2.2 Technical Implementation:

The script uses the pre-trained models to classify example texts and calculates the probability that each text is phishing. These probabilities are then converted into percentages to simplify interpretation. After the calculations, the results are printed in the terminal and matplotlib is used to create bar plots visualizing each model's assessment of the email examples. This method allows for a deeper analysis and comparison of the models' performance and provides valuable insights into their suitability for detecting phishing.

## 3.3 Assessment of Usability and Effectiveness

User tests are conducted using only  logistic regression algorithms to validate the application's usability and effectiveness. Feedback from a user is collected and used for iterative improvements, ensuring that the application meets real user needs.

## 3.4 Tools and Technologies

In addition to Flask used for website development and Python, tools like PyCharm and GitHub have been used for code development and version control, while "ChatGPT 4" plays a crucial role in the project's code optimization and debugging, technical explanation, and troubleshooting. The project benefits from libraries such as spaCy for text analysis, and Scikit-learn, Pandas, as well as Matplotlib for data handling and visualization.

## 3.5 Justification of Method Selection

The choice of NLP and ML is justified by their ability to effectively process and analyze text-based content. Integrating these methods into a user-friendly web application enables a powerful solution for combating phishing and increasing digital security.

# 4 Construction

This project aims to develop a Flask-based web application to effectively distinguish between legitimate and phishing emails. By integrating advanced NLP and ML models, the project aims to create a user-friendly platform that not only identifies phishing attempts but also serves as an educational tool to increase awareness of cybersecurity.

## 4.1 Technical Choices and Methodological Approach

Strategic decisions were made to optimize the system's performance and user experience:

### 4.1.1 Selection of Machine Learning Models:

Logistic Regression, Random Forest, and Gradient Boosting were chosen for their ability to handle the complexity and variations in phishing emails, thanks to their effectiveness in similar classification tasks.

### 4.1.2 Data Preparation and Preprocessing:

A rigorous preprocessing process was introduced to ensure data quality and relevance, involving steps such as normalization, tokenization, and lemmatization, optimizing text data for machine learning models.

### 4.1.3 Integration and Web Application Development:

By leveraging Flask, a dynamic and interactive web application was created, allowing users to classify emails in real-time and receive immediate feedback about potential security threats.

### 4.1.4 Use of ChatGPT for Code Improvement and Dataset Generation:

The project benefited from AI-based tools, such as ChatGPT, for code optimization, technical explanations, and debugging, accelerating the development process and enhancing the system's robustness and efficiency. ChatGPT was also used to generate datasets used in the project.

## 4.2 Analysis and Solution Approach

To manage the project's scope and complexity, both top-down and bottom-up strategies were used:

### 4.2.1 Top-down:

The project was initially divided into larger construction components (e.g., data access, model development, user interface), enabling a structured overview and planning. Then, the report writing and testing began.

### 4.2.2 Bottom-up:

By focusing on specific techniques and solutions, the system could be gradually built, with continuous feedback and adaptation based on test results and user feedback.

## 4.3 Implementation and Optimization

The final phase involved extensive coding, testing, and optimization. By applying cross-validation and hyperparameter optimization, the models' effectiveness could be fine-tuned, resulting in a high-performing and reliable application.

# 5 Results

The results section of the report aims to present the objective findings of a systematic investigation into the email classification system. This section includes outcomes from three machine learning models – Logistic Regression, Random Forest, and Gradient Boosting – to ascertain if email messages are legitimate or phishing attempts.

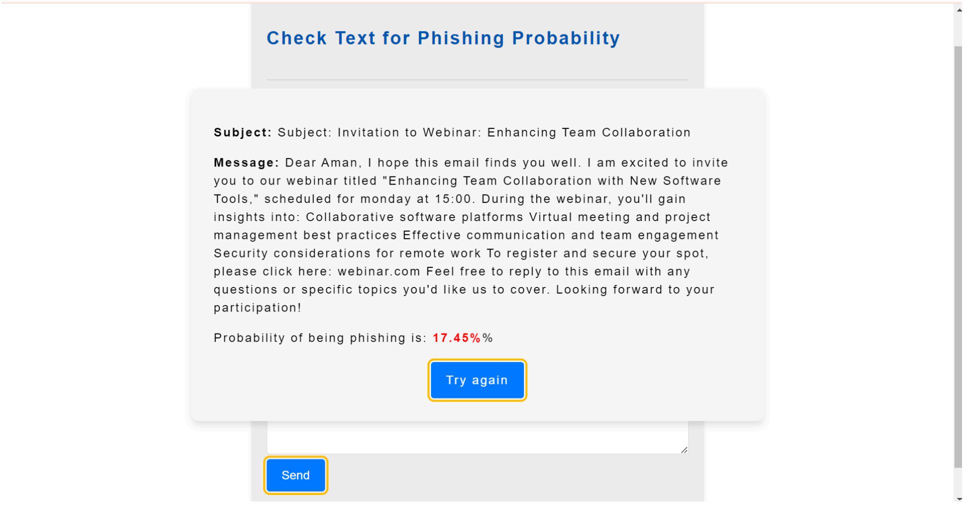
## 5.1 User Test

This part presents the outcomes from two distinct tests carried out by two testers. Each tester performed two trials: the first involved inputting legitimate text into the model for training purposes, and the second focused on inputting a text representing a phishing attempt. The goal was to compare results between the different testers and the two types of text input.

### 5.1.1 Test Results

**Tester 1 legitimate text test:**

Figure 5 shows the results from the first tester, reflecting the estimated probability that the submitted text is authentic. According to these results, the submitted message is assessed to have a 17.45% probability of being fake.



Check Text for Phishing Probability

**Subject:** Subject: Invitation to Webinar: Enhancing Team Collaboration

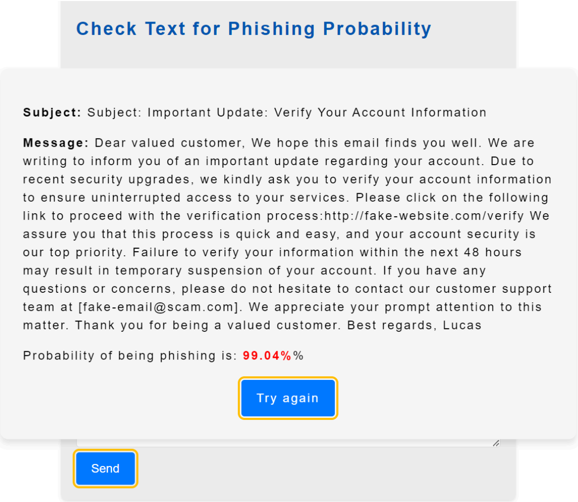
**Message:** Dear Aman, I hope this email finds you well. I am excited to invite you to our webinar titled "Enhancing Team Collaboration with New Software Tools," scheduled for monday at 15:00. During the webinar, you'll gain insights into: Collaborative software platforms Virtual meeting and project management best practices Effective communication and team engagement Security considerations for remote work To register and secure your spot, please click here: webinar.com Feel free to reply to this email with any questions or specific topics you'd like us to cover. Looking forward to your participation!

Probability of being phishing is: 17.45%%

*Figure 5: Tester number 1 results legitimate*

### Tester 1 phishing test:

In the second trial in figure 6 by tester number one, where the tester intentionally input suspicious information, the model's analysis indicates there is a 99.04% probability that the entered message is a phishing attempt.



Check Text for Phishing Probability

**Subject:** Subject: Important Update: Verify Your Account Information

**Message:** Dear valued customer, We hope this email finds you well. We are writing to inform you of an important update regarding your account. Due to recent security upgrades, we kindly ask you to verify your account information to ensure uninterrupted access to your services. Please click on the following link to proceed with the verification process:http://fake-website.com/verify We assure you that this process is quick and easy, and your account security is our top priority. Failure to verify your information within the next 48 hours may result in temporary suspension of your account. If you have any questions or concerns, please do not hesitate to contact our customer support team at [fake-email@scam.com]. We appreciate your prompt attention to this matter. Thank you for being a valued customer. Best regards, Lucas

Probability of being phishing is: **99.04%**%

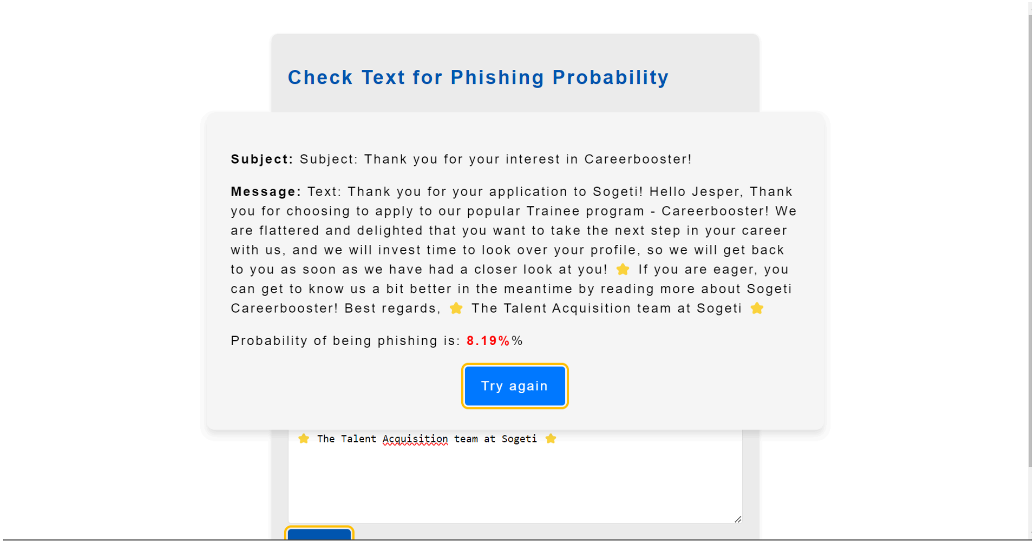
*Figure 6: Phishing attempt shown in the analysis.*

### Tester 1's thoughts on the results and feedback:

It looks promising and good. Implement Random Forest and Gradient Boosting. Choosing a balanced dataset is important for performance.

### Tester 2 legitimate text test:

Figure 7 presents the outcome from the first trial in the second round of analyses by the second tester, indicating there is an 8.19% probability that the submitted text is a phishing attempt.



Check Text for Phishing Probability

**Subject:** Subject: Thank you for your interest in Careerbooster!

**Message:** Text: Thank you for your application to Sogeti! Hello Jesper, Thank you for choosing to apply to our popular Trainee program - Careerbooster! We are flattered and delighted that you want to take the next step in your career with us, and we will invest time to look over your profile, so we will get back to you as soon as we have had a closer look at you! ⭐ If you are eager, you can get to know us a bit better in the meantime by reading more about Sogeti Careerbooster! Best regards, ⭐ The Talent Acquisition team at Sogeti ⭐

Probability of being phishing is: **8.19%**%

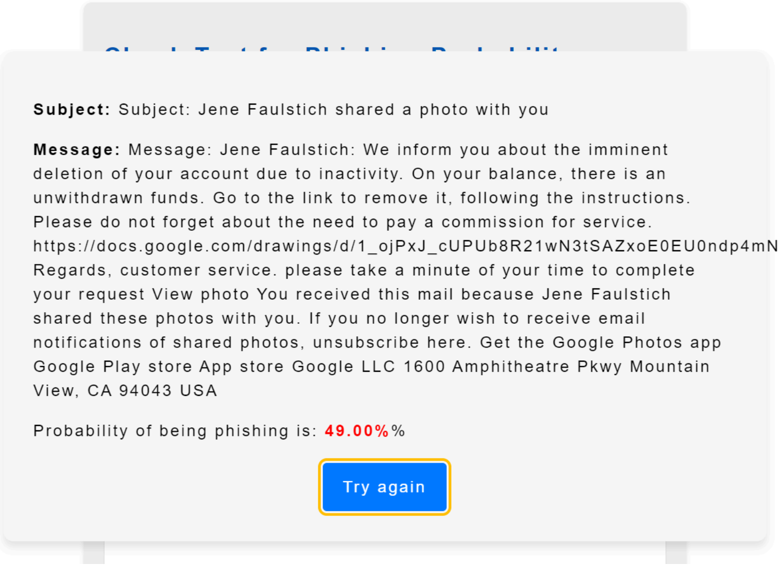
⭐ The Talent Acquisition team at Sogeti ⭐

*Figure 7: Legitimate text input by tester 2.*

## Tester 2 phishing test:

Figure 8 shows the results from the second trial by tester number two, indicating the submitted text has a 49% probability of being a phishing attempt.

*Figure 8: Displays a phishing attempt with a 49% probability, tested by user 2.*



**Subject:** Subject: Jene Faulstich shared a photo with you

**Message:** Message: Jene Faulstich: We inform you about the imminent deletion of your account due to inactivity. On your balance, there is an unwithdrawn funds. Go to the link to remove it, following the instructions. Please do not forget about the need to pay a commission for service. https://docs.google.com/drawings/d/1_ojPxJ_cUPUb8R21wN3tSAZxoE0EU0ndp4mNLJDsyBE/pre
Regards, customer service. please take a minute of your time to complete your request View photo You received this mail because Jene Faulstich shared these photos with you. If you no longer wish to receive email notifications of shared photos, unsubscribe here. Get the Google Photos app Google Play store App store Google LLC 1600 Amphitheatre Pkwy Mountain View, CA 94043 USA

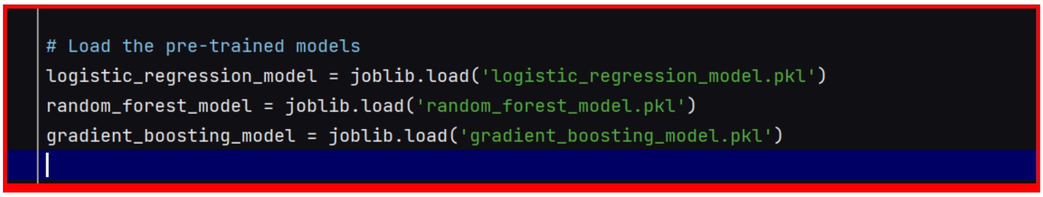
Probability of being phishing is: **49.00%**%

**Tester 2's thoughts on the results and feedback:**

The results look overall positive. However, a critical observation is that phishing texts may succeed in deceiving the AI system due to the dataset being too small. To improve the model's ability to more effectively identify phishing, the implementation of Random Forest and Gradient Boosting techniques is suggested. Additionally, the importance of choosing a balanced dataset to optimize model performance is emphasized.

## 5.2 Comparison of Model Performance

In this section, pre-trained models saved in pkl format are used as shown in Figure 9, which are later used in another Python file to plot them in figures for comparison using matplotlib.



```
# Load the pre-trained models
logistic_regression_model = joblib.load('logistic_regression_model.pkl')
random_forest_model = joblib.load('random_forest_model.pkl')
gradient_boosting_model = joblib.load('gradient_boosting_model.pkl')
```

*Figure 9: Shows how various pre-trained data are loaded and used to plot them in graphs for comparison.*

As shown in Figure 10, a comparison of the three models' assessments for a range of different email messages is presented visually through bar charts. Each chart displays the models' percentage probability values, providing insight into their ability to accurately identify phishing.

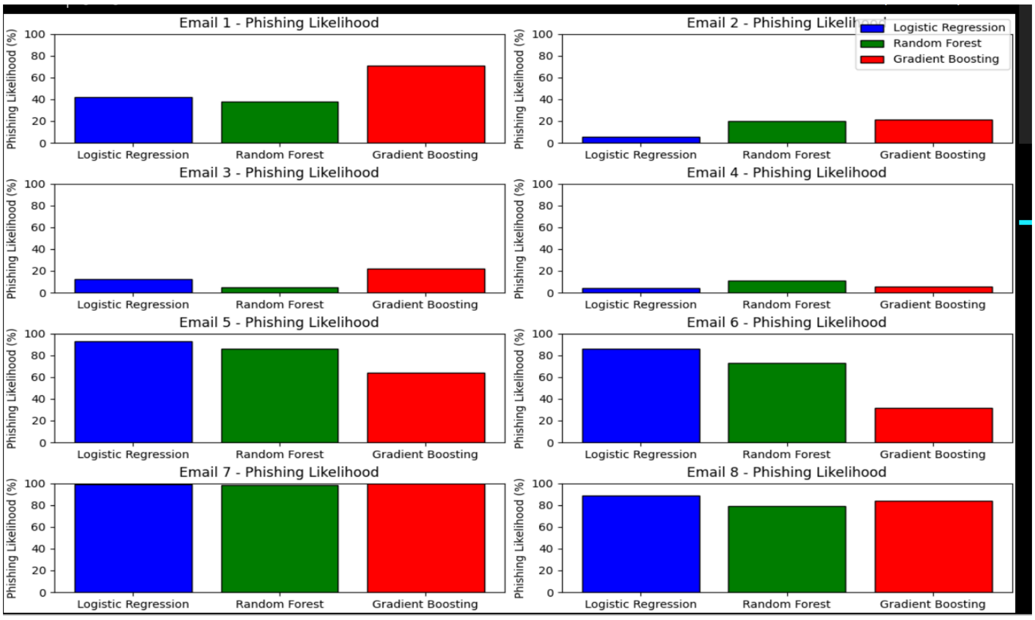
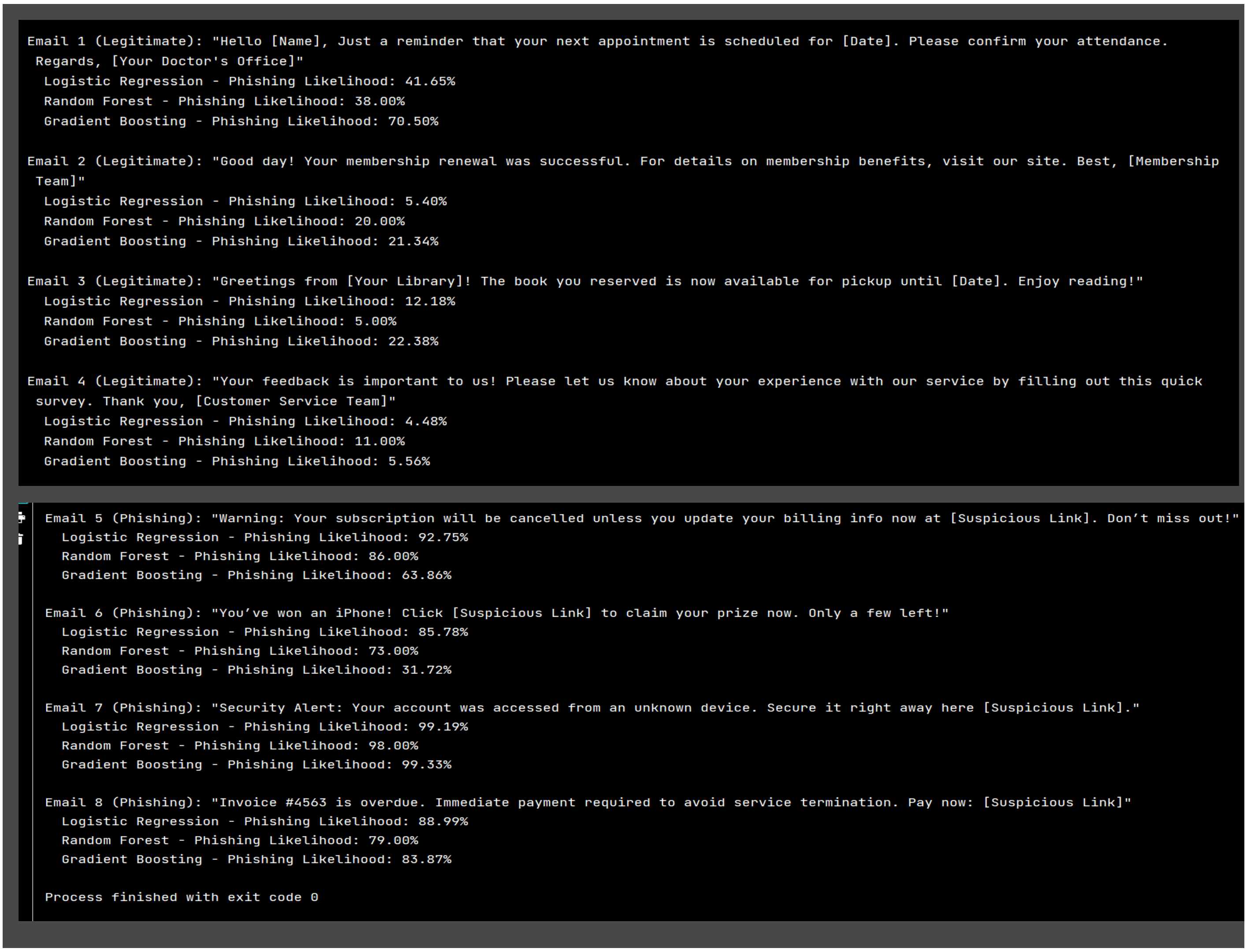


*Figure 10: Comparison of Model Performance*

## 5.3 Terminal Output for Classification

Supplementary to the visual comparison, data have also been gathered through terminal output. This data provides a detailed overview of the models' probability assessments as shown in (Figure 11) for each tested email text. The texts, ranging from potentially legitimate messages to clear phishing attempts, have been assessed by each of the three models to give a comprehensive view of the system's classification capability.



```
Email 1 (Legitimate): "Hello [Name], Just a reminder that your next appointment is scheduled for [Date]. Please confirm your attendance.
 Regards, [Your Doctor's Office]"
  Logistic Regression - Phishing Likelihood: 41.65%
  Random Forest - Phishing Likelihood: 38.00%
  Gradient Boosting - Phishing Likelihood: 70.50%

Email 2 (Legitimate): "Good day! Your membership renewal was successful. For details on membership benefits, visit our site. Best, [Membership
 Team]"
  Logistic Regression - Phishing Likelihood: 5.40%
  Random Forest - Phishing Likelihood: 20.00%
  Gradient Boosting - Phishing Likelihood: 21.34%

Email 3 (Legitimate): "Greetings from [Your Library]! The book you reserved is now available for pickup until [Date]. Enjoy reading!"
  Logistic Regression - Phishing Likelihood: 12.18%
  Random Forest - Phishing Likelihood: 5.00%
  Gradient Boosting - Phishing Likelihood: 22.38%

Email 4 (Legitimate): "Your feedback is important to us! Please let us know about your experience with our service by filling out this quick
 survey. Thank you, [Customer Service Team]"
  Logistic Regression - Phishing Likelihood: 4.48%
  Random Forest - Phishing Likelihood: 11.00%
  Gradient Boosting - Phishing Likelihood: 5.56%
```

```
Email 5 (Phishing): "Warning: Your subscription will be cancelled unless you update your billing info now at [Suspicious Link]. Don’t miss out!"
  Logistic Regression - Phishing Likelihood: 92.75%
  Random Forest - Phishing Likelihood: 86.00%
  Gradient Boosting - Phishing Likelihood: 63.86%

Email 6 (Phishing): "You’ve won an iPhone! Click [Suspicious Link] to claim your prize now. Only a few left!"
  Logistic Regression - Phishing Likelihood: 85.78%
  Random Forest - Phishing Likelihood: 73.00%
  Gradient Boosting - Phishing Likelihood: 31.72%

Email 7 (Phishing): "Security Alert: Your account was accessed from an unknown device. Secure it right away here [Suspicious Link]."
  Logistic Regression - Phishing Likelihood: 99.19%
  Random Forest - Phishing Likelihood: 98.00%
  Gradient Boosting - Phishing Likelihood: 99.33%

Email 8 (Phishing): "Invoice #4563 is overdue. Immediate payment required to avoid service termination. Pay now: [Suspicious Link]"
  Logistic Regression - Phishing Likelihood: 88.99%
  Random Forest - Phishing Likelihood: 79.00%
  Gradient Boosting - Phishing Likelihood: 83.87%

Process finished with exit code 0
```

*Figure 11: Terminal Output for Classification*

# 6 Discussion

## 6.1 Analysis of Model Performance and Methodological Adjustments

The project extensively investigated the performance of different machine learning models to pinpoint the most effective tools for combating cyber threats originating from emails. Initially, simple logistic regression was implemented as a baseline model to establish performance metrics. Subsequently, the analysis transitioned to more advanced and structurally complex machine learning algorithms, including RandomForestClassifier and GradientBoostingClassifier. This gradual progression from basic to advanced methods underscores the importance of adaptability in model and methodology selection, crucial for effectively addressing the dynamics and diversity of cyber threats.

Furthermore, peer feedback played a crucial role in guiding methodological development. Through regular evaluations and discussions, the project was able to benefit from external perspectives, leading to new insights and improvements in the use of machine learning techniques to identify and counter phishing attacks. This process of open feedback has been instrumental in refining the design and functionality of the models.

The analysis of model performance, as evidenced by the quantitative measures of probability assessments presented in Figures 10 and 11, revealed a tendency towards overfitting, particularly in the GradientBoostingClassifier model. This observation is significant as it indicates a risk of the model becoming too specialized to the training data, thereby losing its ability to generalize to new, unseen datasets. Overfitting poses a central challenge in machine learning and demands meticulous consideration during the development of models.

By continuously improving model and methodological choices, the project confirms the importance of an iterative and responsive research process in the development of effective tools against cyber threats. Progress towards models resilient to overfitting and flexible enough to adapt to the evolution of threats constitutes the core of next-generation cybersecurity defense.

## 6.2 Managing Overfitting and Prototype Development

In the section on managing overfitting and prototype development, it emerges that despite its powerful ability to distinguish between legitimate and deceptive communication, the GradientBoostingClassifier shows a tendency to become too finely tuned to the training data. This trend towards overfitting underscores the urgent need to implement more comprehensive testing and validation mechanisms, as well as potentially expanding the size of the datasets used for training. Such a strategy would not only help the model retain its ability to generalize over new, unseen data but also strengthen its robustness against the constantly changing and increasingly sophisticated cyber threats.

This insight raises an important discussion about the need to develop models that are both adaptive and resilient, capable of adjusting to the continuously shifting landscape of cyber threats. Creating such dynamic models allows us to not only respond to today's threats but also anticipate and prevent future attacks.

## 6.4 Increased Awareness and Ethical Considerations

The project highlights the urgent need to educate society about the dangers of cybersecurity. By creating a web-based application that allows users to immediately experience and witness the effects of detecting phishing, the project takes a crucial step towards making sophisticated cybersecurity tools more accessible to the public. This effort can not only strengthen individuals' skills in detecting and sidestepping phishing attempts but also contribute to a broader understanding of the crucial role of cybersecurity in our digital lives. The project has also carefully considered ethical aspects and strived for social responsibility, with a focus on creating transparent and fair systems that respect and protect users' privacy and individual rights.

# 7 Future Research Directions and Concluding Reflections

The project contributes to the field of cybersecurity by demonstrating the value of integrating NLP and ML to enhance the identification of phishing attacks. By combining these two powerful technologies, the project has succeeded in creating a more efficient and rapid method of detecting and responding to fraudulent attempts to compromise digital information. The collaborative utilization of NLP and machine learning has facilitated the creation of advanced algorithms that possess the ability to analyze and comprehend text in a manner resembling human judgment. This capability is essential for detecting nuanced indicators of phishing attempts.

Finally, the project highlights a clear need for continued research in areas aimed at enhancing the flexibility and resilience of models to withstand the challenges of overfitting. Future research efforts should also focus on creating user-friendly cybersecurity solutions accessible to a wide range of users, a critical factor in the fight against advanced cyber threats. The results of this project lay a solid foundation for future explorations in cybersecurity and machine learning while also encouraging continued discourse on the most effective methods to protect our digital communities against ever-evolving threats.

# References

[1] Upadhyay A. Detecting Phishing Attacks with AI Medium. Accredian; 2023 Oct 23 [cited year month day]. Available from: https://medium.com/international-school-of-ai-data-science/phishing-detection-met-generative-ai-365b3e89920d Accessed March 20, 2024.

[2] Benavides-Astudillo E, Fuertes W, Sanchez-Gordon S, Nuñez-Agurto D, Rodríguez-Galán G. A phishing-attack-detection model using natural language processing and deep learning. Appl Sci. Available from: https://www.mdpi.com/2076-3417/13/9/527 5 Accessed March 24, 2024

[3] MSB, ”Cybersäkerhet i Sverige– Hot, metoder, brister och beroenden”, https://www.msb.se/siteassets/dokument/amnesomraden/informationssakerhet-cybersakerhet-och-sakra-kommunikationer/nationellt-center-for-cybersakerhet/rapport-cybersakerhet-i-sverige-2020--hot-metoder-brister-och-beroenden.pdf . Published 2020. Accessed March 15, 2024.

[4] Yang, Rundong et al., ”Social Engineering Attack-Defense Strategies Based on Reinforcement Learning”, Tech Science Press, 2023, https://www.techscience.com/csse/v47n2/53636/pdf. Published 2023. Accessed March 15, 2024.

[5] IBM. “What is Natural Language Processing? NLP”. IBM; https://www.ibm.com/topics/natural-language-processing Accessed March 16, 2024.

[6] BM. “What Is Machine Learning (ML)?”. IBM; https://www.ibm.com/topics/machine-learning Accessed March 16, 2024.

[7] Ding, Yan, Luktarhan, Nurbol, Li, Keqin, & Slamu, Wushour, ”A keyword-based combination approach for detecting phishing webpages”, Computers & Security, 84, 256-275, https://doi.org/10.1016/j.cose.2019.03.018 . Published 2019. Accessed March 16, 2024.

[8] IBM. “What Is Random Forest?”. https://www.ibm.com/topics/random-forest Accessed March 16, 2024.

[9] Subasi, Abdulhamit, Molah, Esraa, Almkallawi, Fatin, & Chaudhery, Touseef J., ”Intelligent Phishing Website Detection using Random Forest Classifier”, 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), 1-5 November,2017.https://ieeexplore.ieee.org/abstract/document/8252051/ . Accessed March 16, 2024.

[10] Omari, Kamal, ”Phishing Detection using Gradient Boosting Classifier”, Procedia Computer Science, Vol. 230, sidorna 120–127, 2023, https://doi.org/10.1016/j.procs.2023.12.067. 3rd International Conference on Evolutionary Computing and Mobile Sustainable Networks (ICECMSN 2023). Published 2023. Accessed March 16, 2024.

[11] Sheridan, P., & Onsjö, M. (2024). “The hypergeometric test performs comparably to TF-IDF on standard text analysis tasks.” *Multimedia Tools and Applications: An International Journal*, 83(10), 28875-28890. Accessed March 16, 2024.

[12] Soffos AI. Top 8 Python Libraries For Natural Language Processing (NLP) in 2023. Medium. 2023 Aug 22. Available from: https://medium.com/@soffosdotai/top-8-python-libraries-for-natural-language-processing-nlp-in-2023-5963bfa53296 Accessed March 20, 2024.

[13] Santiago D. Balancing Imbalanced Data: Undersampling and Oversampling Techniques in Pytho. Medium. 2023 Jun 5 https://medium.com/@daniele.santiago/balancing-imbalanced-data-undersampling-and-oversampling-techniques-in-python-7c5378282290 Accessed March 20, 2024.

.

# Appendix A:

**Source:**

https://github.com/KORAY-AMAN-ASLAN/cource_softwareSecurity