IJEMD-CSAI, Special Issue (2026) https://doi.org/10.54938/ijemdcsai.2026.04.2.665

International Journal of Emerging Multidisciplinaries: Computer Science and Artificial Intelligence

Research Paper

Journal Homepage: www.ojs.ijemd.com
ISSN (print): 2791-0164 ISSN (online): 2957-5036

An Explainable Multi-Modal Phishing Detection Framework

Wong Ki Hurn [1], Tan Yi Yan [1], Dickson Lim Zi Yuan [1], Li Qian [1], Tee Jim Host [1], Zhang Xiaolong [1], Yip Hong Seng [1], Siva Raja Sindiramutty [1]

1. School of Computer Science, Taylor's University, Subang Jaya, 47500, Selangor Malaysia

1.0 Abstract

Phishing is one of the most common online threats, and it is becoming harder to detect because attackers use AI nowadays, fake website designs, and new tricks. Traditional methods like simple rules or URL reputation checks are no longer enough to stop these modern attacks. This report introduces Phishing Shield AI, a phishing detection system that uses three types of analysis: text analysis with NLP, URL and domain checking, and computer vision to compare webpage visuals. Each part gives a risk score, and the system combines them to decide whether something is phishing. It also provides clear explanations by showing which words, links, or images look suspicious. Based on the literature review, gap analysis, system design, and evaluation, the multi-modal approach improves accuracy, reduces mistakes, and adapts better to new phishing methods. Overall, Phishing Shield AI is a practical and scalable solution that performs better than single-method systems and has strong potential for future improvement.

Keywords:

Multi-Modal Phishing Detection, Artificial Intelligence in Cybersecurity, Natural Language Processing (NLP), URL and Structural Feature Analysis, Explainable AI (XAI)

Email Addresses : 0364414@sd.taylors.edu.my, 0364652@sd.taylors.edu.my, 0382704@sd.taylors.edu.my, 0375290@sd.taylors.edu.my, 0376783@sd.taylors.edu.my, 0368771@sd.taylors.edu.my, 0364391@sd.taylors.edu.my

2.0 Introduction & Problem Identification

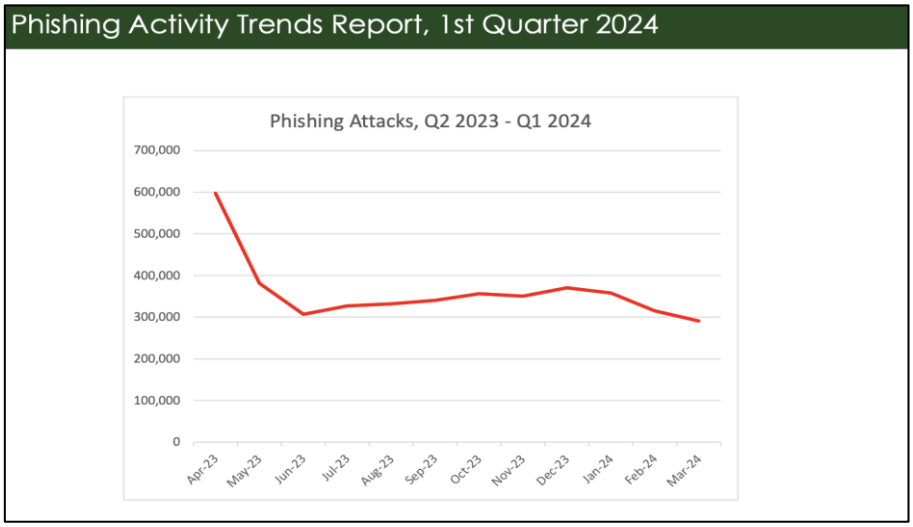

Figure 1.0 Phishing attacks, Q2-2023- Q1 2024 (Anti-Phishing Working Group, 2025).
What is an AI-powered phishing detection system? Before this, the definition of phishing needs to be introduced to gain a deeper understanding about it. Phishing is a crime employing both social engineering and technical subterfuge to steal consumers' personal identity data and financial account credentials (Anti-Phishing Working Group, 2025). When users install or visit a malicious website, they might get deceived into revealing sensitive information. The human emotions like trust and sense of urgency make phishing become one of the most persistent cyber threats in today's digital world (Sangfor Technologies, 2022; Riskhan et al., 2025).

According to the Anti-Phishing Working Group, phishing activity remained consistently high throughout 2023 and into the first quarter of 2024, with only modest fluctuations in overall attack volume. As shown in Figure 1.0, the number of detected phishing attacks decrease greatly from 600,000 to 300,000 in the mid of 2023. This trend shows that while some short-term mitigation efforts have reduced immediate peaks, phishing continues to persist at hundreds of thousands of incidents per quarter. The sustained baseline indicates the adaptive nature of phishing operations and shows the urgent need for advanced detection mechanisms to end this cyber threat. For example, AI-powered phishing detection systems are capable of this.

Back to the topic, Artificial Intelligence (AI) has emerged as a promising tool in the fight against phishing. AI-driven phishing detection systems leverage machine learning, natural language processing (NLP), computer vision, and other AI approaches to identify those potential phishing activities in emails or websites. The AI technique, like the machine learning model, forms the foundation of the system, it learns from large datasets of legitimate and phishing emails to recognize the differences and makes predictions about phishing attacks. Other techniques like computer vision are visually inspecting images used in webpages or emails and compare them with known figures to detect impersonation or logo misuse.

Furthermore, NLP is also applied to analyse the textual content of emails or webpages that contains social-engineering cues such as urgency, authority, or requests for confidential information. NLP extracts the phishing information from raw text data through entity recognition, relationship extraction, semantic understanding, contextual analysis, and degree of certainty. For instance, NLP identifies entities like names, brands, URLs, or financial terms within the text. As phishing emails often misuse legitimate entities to gain trust. Next, NLP extract the relationship exists in texts, if there is a relationship that need user to enter sensitive information may considered as phishing attempt. NLP also interprets meaning beyond keywords to check is there any suspicious action. The tone and writing style within the context will be evaluated to detect inconsistencies. Lastly, NLP model will need to evaluate the confidence about its interpretations or extracted information in labelling the content as phishing or benign.

There are many solutions to fight against the phishing emails or websites. However, the most common solution for phishing emails and websites are heuristic analysis and visual similarity detection respectively. So, what is a heuristic method first? It is a method that used to detect suspicious behaviour in emails by using rules and algorithms. The method can identify phishing patterns such as suspicious links, malicious email addresses, specific keywords, and so on. Heuristic analysis with the implementation of AI will become more efficient in pattern matching and improve the predefined rules (Hornetsecurity, n.d.; Keon et al., 2025). While heuristic-based filtering is efficient and easy to implement, it exhibits significant limitations. The dependence on static crafted rules makes it ineffective against changing phishing strategy, such as phishing attacks that using new wording, techniques, or generated links which is unpredictable. Besides that, this method also suffers from high false positive and false negative rates. For example, legitimate messages may be incorrectly considered as phishing attacks, while the real attacks remain undetected. This is because of the method lack semantic understanding, it can recognize patterns but fail to understand the meaning beyond the words.

Another widely adopted approach, particularly for phishing websites, is visual similarity detection. This approach employs computer vision techniques to compare the figures of a webpages with existing legitimate images, such as from the perspective of layout, colour scheme, logos, and so on. Although visual similarity detection is effective in identifying clone websites, it still very expensive and resource-intensive because the webpage needs to be analysed in real time. The attackers can simply escape the detection by just doing a small change, such as slightly shifting logos or modifying colours to alter appearance upon inspection. In order to maintain and update the library of legitimate website references, the maintenance fee is going to be very high. Thus, the effectiveness of result matching is not matching its expensive cost. This is why visual similarity detection alone is inadequate to provide adaptive protection against the evolving landscape of phishing websites.

The example of real-world incidents of phishing attack is on December 28, 2023, a phishing attacks by the Russia-linked state-sponsored APT28 group also known as Fancy Bear against Ukrainian government agencies and several organizations in Poland. Attackers sent phishing emails containing links which first led to malicious files, eventually infecting victims with malware. The phishing emails looks trustable because of using social engineering which is a manipulation technique that exploits human error to fool people giving their private or sensitive information without authentication (Telychko, 2023).

The email links redirect targeted users to a crafted phishing website that leads to downloading a shortcut file via JavaScript and the "search" ("ms-search") application protocol. When the user opened the file, it triggered a PowerShell command to secretly downloaded and executed additional malware components from a remote server. Within an hour from the moment of the initial click, the infected systems began communicating with attackers' command and control servers. This allows APT28 to install backdoors like OCEANMAP and tools were used to steal and exfiltrate sensitive data (Telychko, 2023; Kiyani et al., 2024).

According to the MITRE ATT&CK framework, this attack maps to the technique T1566 – Phishing, more specifically the sub-technique T1566.002 – Spearphishing Link. This is because the incidents match the definition of the technique, which is spearphishing emails with malicious link is sent to try to access to victim systems and exploit sensitive data (MITRE ATT&CK® , 2020).

This real-world incident shows how modern phishing campaigns are multi-stage and highly adaptive. The organizations cannot longer rely solely on static, rule-driven detection. From this incident, it highlights the urgent need for AI-powered phishing detection systems that can combine machine learning, NLP, and computer vision approaches to detect deceptive patterns accurately. The ultimate lesson that can be learned from this case is only adaptive, intelligent, and multi-modal detection systems can fight against the evolving landscape of phishing attacks.

3.0 Literature Review & Gap Analysis

3.1 NLP Approaches to Phishing Email Detection

The two main methods to detect phishing emails are Term Frequency-Inverse Document Frequency (TF-IDF) and word embeddings. TF-IDF tells us how important a word is in one document compared to many other documents. The most used machine learning algorithm was Support Vector Machine (SVM). It is a machine learning algorithm that is used for classification and regression tasks. (Salloum et al., 2022)

A survey found that NLP techniques targeting text content such as subject lines, message body and sender text that can capture clues of social engineering. These clues include urgent requests, unusual syntax, or context mismatch. (Salloum et al., 2021) Deep-learning models (DL) and transformer-based text models have been applied recently. DL models are better than traditional machine learning (ML) models in detecting phishing emails between multiple datasets. (Alhuzali et al., 2025) This is because DL models can automatically learn features from raw data. The process does not need much manual feature engineering compared to ML models. (Chris, Johnson and Phonix, 2024)

NLP-based detection is beneficial because it can detect phishing emails that may have clean URLs or links that look legit but are still fake due to social engineering text. It scans the emails in milliseconds. This helps to block threats before they reach anyone's inbox. (Linux Security, 2025) However, many studies are limited to English datasets, feature engineering remains labour-intensive and models often assume that the data is static. Scammers now are free to use any type of natural language generation tools to create phishing emails that make it easy for people to easily believe their content.

3.2 Multi-Modal Approaches for Phishing Websites and Emails

There are still many people who have been scammed not only by phishing emails but also by websites. Therefore, the visual or structural size of phishing websites is receiving more and more attention. Phishers often try to copy the look, fonts, logos, layout or images in emails of legitimate websites to bypass text-based filters.

To prevent these tricks, methods that analyse HTML/DOM features, URL structure have been developed. In some cases, visual features can also be obtained through computer vision. Machine-learning methods using URL, HTML source-code and image-features can achieve high accuracy and have a lower false positive rate than other systems that only have one type of feature. (Tang and Mahmoud, 2021; Al-Musib et al., 2021). Phishing websites often share the same structural and visual features that can be detected through multimodal analysis.

Besides that, computer vision has become a powerful method for detecting phishing websites. VisualPhishNet (Abdelnabi, Krombholz and Fritz, 2020) uses a convolutional neural network (CNN) to compare webpage screenshots with other known legitimate pages. If the visual is similar in an unusually high level, the system will mark the page as suspicious. This shows that phishing websites often rely heavily on visual imitation.

These structural/visual and multi-modal techniques are useful because they can catch phishing attempts that might go through text-based filters. By analyzing the visual similarities or unusual structural features of websites, detection systems can block more complex attacks.

3.3 Overview of Research Results

AI-powered phishing detection systems integrating NLP and structural/visual components have made good progress. Methods that have multiple classifiers, deep text learning, website structure features and multimodal fusion is currently the most advanced technique to detect phishing emails and websites. These techniques can improve detection accuracy, address data imbalance in context and are often smarter than single models. (Spring Nature Link, 2024; Aldughayfiq et al., 2023)

There are challenges that still need to be addressed. Most phishing detection models produce false positives. Legitimate websites or emails are incorrectly detected as phishing attacks. Deep learning models are more vulnerable to attacks because they rely on the recognition of subtle patterns. Scaling up the phishing detection systems to real-time operation is also a major challenge. Dataset imbalance that causes the model to overfit to legitimate examples and fail to generalize to new phishing attacks is also a part that needs to be solved. (Spring Nature Link, 2024)

3.4 Gap Analysis

First, the gap is that most phishing detection systems are not tested in real-world environments. Most systems are only tested using old or static datasets that can be controlled in lab environments. In real life, phishing attacks will not stop developing. So, outdated datasets will no longer be able to deal with advanced phishing attacks. Attackers create new types of emails, new website designs, and even use

artificial intelligence to generate fake messages that make everyone look professional and believe the information is legitimate. Models that work well in labs may fail in real internet scenarios. (Team, 2025; Alwakid et al., 2023) So, detection systems must be tested and updated regularly using real world rather than using static samples.

Another gap is language barrier. Most phishing detection systems only perform well in English language content. But, phishing attacks also come in different languages. Attackers can use translation applications to translate spam messages into different languages they want. People who did not learn English before will face a high risk of being scammed. A study compared the accuracy using different machine learning methods to filter English and Arabic spam messages. The result showed that the performance to filter Arabic messages is more inaccurate. (Alaa El-Halees, 2009) Therefore, a phishing detection system needs to understand and analyze phishing content in different languages.

It is hard to detect AI-generated phishing content. With a few commands, attackers can use tools like ChatGPT to create phishing contents that look highly professional and have no spelling or grammar errors. Old filtering methods can no longer detect this content. AI-generated phishing emails often reach their targets without being detected by existing systems. (Brissett and Wall, 2025; Ghosh et al., 2022) This means that detection systems now must have the ability to detect AI-generated phishing content.

Another gap is lack of explainability in many AI-based phishing systems. Some models can detect phishing with high accuracy but it would not tell users how they make their decisions. (Arxiv.org, 2022) This will let users confused about the reason. Users need to know why related emails or websites have been identified as phishing emails. Systems that lack explanation and transparency are difficult to trust or improve.

Overall, there are still major gaps existing. These gaps provide strong opportunities for developing better, smarter, and more practical phishing detection systems.

4.0 Proposed Secure System

4.1 System Name and Core Concept

Phishing Shield AI is a multi-modal AI anti-phishing detection tool. With the integration of NLP, compute vision, and structural feature analysis, it can identify email and website phishing. This tool aims to make up for the shortcomings of many existing phishing detection systems, including improving detection accuracy, reducing false positives, providing detection basis, while supporting automatic learning and real-time update functions.

4.2 System Architecture

The Phishing Shield AI system uses a modular and scalable system. From Figure 2.0, the phishing detection will go through six primary systems:

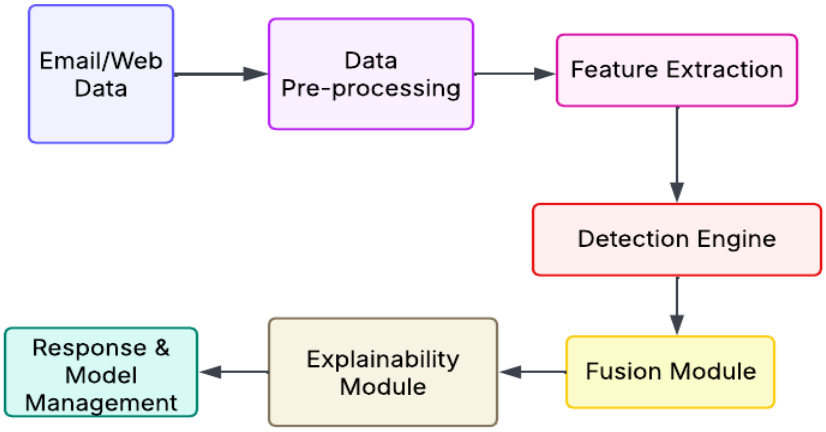

Figure 2.0 Phishing Shield AI's Architecture Diagram

1. Data Collection and Pre-processing

This module gathers and compiles incoming emails and website access requests and user-reported links. It analyzes and processes the emails, extracting subjects and bodies, and links to attachments and images, and converts web pages into structured text or screenshots and prepares them for further analysis.

2. Feature Extraction

This step extracts several types of key information from the content to be detected, including text-related features (such as key words in the content, the overall tone of the writing, and specific names mentioned, such as the names of institutions or individuals); structure-related features (covering the existence time of the domain name, the length of the link, and the number of redirects when accessing the page); and visual-related features (mainly referring to the logo elements on the page and the similarity between the page layout and that of regular official pages).

3. Multimodal Detection Engine

This module forms the backbone of the service and grades them by risk. It combines the following three different analysis models :

Natural Language Processing (NLP) Analysis: focuses on analyzing the text content and language structure in emails or webpages. Its main goal is to identify all the expressions that are used to manipulate users' thinking or intentionally make users feel urgent or emotionally upset.Through deep semantic analysis, the system accurately detects typical Social Engineering cues, such such as abnormal requests for sensitive information, exaggerated reward claims, or sharply worded warnings, allowing it to determine malicious intent.

Structure and Metadata Analysis (XGBoost): This tool focuses on verifying technical aspects and non-content data. It does not analyze the text itself, but evaluates the underlying technical indicators supporting emails or links, including structured features such as domain registration duration, link length, number of redirects, and sender metadata. By identifying anomalies in these indicators, the tool

can quickly pinpoint suspicious network architectures, providing a key basis for overall risk assessment.

Visual Image Analysis (CNN/ViT): This model utilizes advanced computer vision technology to detect visual mimicry in phishing pages.It carries out a detailed and meticulous analysis of a potentially harmful web page's screenshot by visually comparing it to known representations of legitimate brands. If a screenshot provides a high similarity score the page is likely to clearly and purposefully attempt to impersonate trusted brands and schemes to try to earn user trust.

4. Fusion Module

Integrates scores from multimodal detection engine using a small neural network to produce the final phishing probability.

5. Explainability and Evidence Generation

Highlight suspicious expressions and display the risk characteristics of links.

6. Response and Model Management

Considering the phishing probability, the system will decide to block, quarantine, or flag the content. Analyst feedback is captured to retrain models for ongoing refinements. Such layered design ensures the system remains scalable and enough to adapt to evolving phishing tactics.

4.3 Technical Design & Implementation Details

4.3.1 Overview of System Operation

The Phishing Shield AI system utilizes a multi-step plan for identifying and responding to phishing attempts. To begin with, it collects and retrieves key information such as subject lines, messages, URLs, and screenshots from the Phishing Shield AI system. Incoming emails and website requests are processed. These data are then converted into organized information that the AI models can process. Each model, whether text-based, structural, or visual analyzes the data on its own and assigns a risk score. A fusion layer combines all these results to reach a final decision. If a message is identified as phishing, the system automatically blocks or isolates it and saves the related evidence to help the system learn better in the future.

4.3.2 Core Components and Algorithms

NLP Module

This section of the paper addresses the analysis of the text content of emails. It explains the use of transformer architecture models (like DistilBERT) that specialize in identifying the language patterns concerning social engineering. It elaborates how the model processes the subject line and the body of the email to understand context and extract signals around the email. This includes identifying the signals of urgency, requests for sensitive information, or an abnormal tone. It outputs a probability score showing how likely the text is phishing.

Structural Module

The structural model examines technical attributes such as domain age, URL length, IP-in-URL, number of redirects, and sender reputation. This approach employs a gradient-boosted tree algorithm (XGBoost) as it handles both numerical and categorical features exceptionally well. This module is efficient and quick, which enables real-time URL inspection.

Vision Module

The vision model inspects images or webpage screenshots using a convolutional neural network (CNN). It compares the appearance of the page with known legitimate brand templates.

Fusion Module

The fusion layer combines the prediction scores from all three models. A small feed-forward neural work is used to learn how to weigh each model's result, producing a final phishing probability. This fusion helps to balance the strengths of each model and reduce false positives.

4.3.3 Data Processing and Model Training

Each model is trained on datasets collected from public phishing repositories such as PhishTank, as well as legitimate datasets like the Enron email corpus. Before training, all text data are cleaned, tokenized, and labeled as either "phishing" or "benign." For structural data, URL and domain features are extracted and normalized. Visual training data consist of screenshots of legitimate and phishing websites.

Each model is trained on its own and assessed using metrics like Accuracy, Precision, Recall, and F1-score. After being put into use, Phishing Shield AI keeps gathering newly labelled data through user feedback. It also updates its models regularly to keep up with new phishing tactics.

4.3.4 Detection Workflow

```
def detect_phishing(email):
    data = preprocess(email)
    text_score = nlp_model.predict(data['text'])
    struct_score = struct_model.predict(data['url'])
    if text_score > 0.6 or struct_score > 0.6:
        visual_score = vision_model.predict(data['screenshot'])
    else:
        visual_score = 0
    final_score = fusion_model.combine(text_score, struct_score, visual_score)
    if final_score > 0.8:
        action = "block"
    else:
        action = "allow"
    log_result(email, final_score, action)
    return action
```

Figure 2.1 Phishing Shield AI Detection Algorithm

From Figure 2.1, the pseudocode shows the main detection process. The system first used the text and structural models for a quick check. If the message looks suspicious, the vision model runs for deeper analysis.

4.4 Prototype Design & Simulation Plan

4.4.1 Prototype Design

The prototype of Phishing Shield AI is designed to simulate a real-time phishing detection interface. It allows users to input emails, URLs, or screenshots to check for phishing characteristics. The system integrates natural language, visual, and URL analysis modules to ensure a holistic inspection.

The user interface (UI) prototype is simple and user-friendly. It includes three major components:

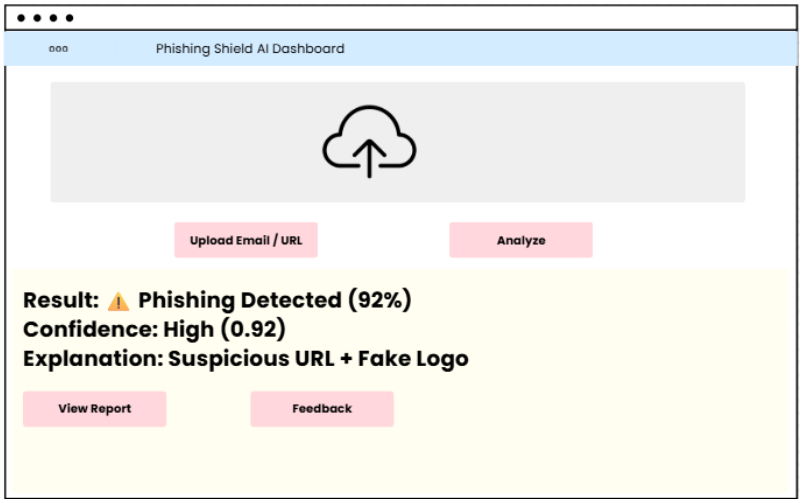

Figure 2.2 Phishing Shield AI Prototype Design

1. Input Panel – where users can upload email text or URLs.
2. Detection Panel – displays real-time analysis status and progress.
3. Result Dashboard – provides detection results, risk levels, and recommended user actions.

4.4.2 Data and Tools

Phishing URLs were gathered from PhishTank and authentic emails were extracted from the Enron Email Corpus in order to construct the prototype. To help the system distinguish between safe messages and phishing attempts, the collected data was meticulously cleaned and labelled. XGBoost was utilized to train the machine learning model, and Kaggle assisted in managing the datasets for the Python prototype. Additional NLP and image processing tools were used to process screenshots, check URLs, and analyse email content in order to strengthen the system and give it more ways to correctly identify phishing threats.

4.4.3 Simulation Plan

The simulation process begins with collecting and labelling both phishing and legitimate data. Once prepared, the data is pre-processed and features are extracted for analysis. The dataset is then divided into 70% for training the system and 30% for testing its performance. Results generated by the NLP, Vision, and URL analysis modules are combined in a Fusion Module, which determines whether the input is safe or suspicious. Finally, the system's accuracy and reliability are measured using evaluation metrics such as accuracy, precision, recall, and F1-score.

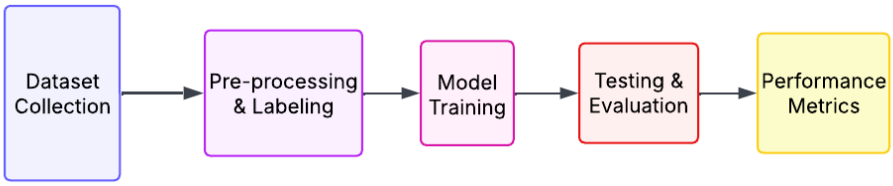

Figure 2.3 Phishing Shield AI Simulation Workflow

4.5 Implementation Challenges & Feasibility

There are some difficulties that are both technical and practical in implementing Phishing Shield AI. The synthesis of different AI models like NLP, structural analysis, and computer vision detection frameworks can become convoluted. Each module has varying input types and preprocessing methods, which may generate issues of incompatibility and detection delays in real time. Data and computational resource constraints are other significant difficulties. Phishing is an ever-changing target and requires constant adaptations of the training dataset. Gathering authentic, phishing samples in multiple languages is a challenge. Model training and inference being computationally expensive is a reality, particularly in smaller organizations and research efforts due to cloud GPU rentals.

There is no doubt that the system, in spite of all that, is operational. The principal smart tools needed are already freely accessible in open-source repositories. Using cloud platforms like AWS and Azure, the system can grow and get real-time updates. Because of the modular architecture of the system, each component can be upgraded separately. The system is being developed with the goals of improving multilingual support, speeding up detection for instant protection, and creating user feedback channels to help the AI learn continuously.

4.6 Why Phishing Shield AI is Superior

Phishing Shield AI works in more ways than normal systems. It checks text, website structure, and images at the same time. Other systems only look at one part, so they miss many attacks. This system also shows users what is risky and why. Users can see which words, links, or pictures look suspicious. The system learns from user feedback to get better every day. It can be used easily on cloud platforms like AWS or Azure. Each part of the system can run alone or together. This makes it faster and easier to update.

4.7 Technical Specifications

4.7.1 System Specifications

The system uses Python 3.10. For model training and image checking, the system runs with TensorFlow, PyTorch, Scikit-learn, and OpenCV. MySQL or MongoDB is used for the database. It stores phishing data, results, and user feedback. These include DistilBERT for text, XGBoost for structure, and CNN or ViT for images. A small neural network joins all the results. The dataset was retrieved from PhishTank, the Enron Corpus, and Kaggle. They provide real phishing and safe samples

to enable learning. Cloud platforms-such as AWS EC2 and Azure ML-can be used to develop the systems. This enables the quick training of models and processing of a lot of data. Streamlit can be used to create the user interface in order to make the process of testing e-mails and links very easy for users. In this regard, the system will use HTTPS for protection against data breaches and SHA-256 hashing to protect user information. Each of these parts connects together to ensure the working of all these segments.

4.7.2 Algorithm Workflow

The Phishing Shield AI system follows a simple step-by-step process in analysing phishing activities. In this regard, the system starts by taking email, website links, or screenshots as input. Three sorts of data are further extracted: text, structure, and visual information. Each part of the system checks one sort of data. The NLP model reads and identifies the text content. The XGBoost model checks the structure, like a domain name, URL length, and redirects. The CNN or ViT model examines the image or screenshot with a focus on visual similarities with the known phishing pages. Indeed, each of these models generates a score representing the probability of the content being classed as phishing. Thereafter, the system combines all the scores into one final phishing score. This score will help the system make a decision on what to do with the message. If the score is high, it blocks the message; if medium, it quarantines it; and if low, it allows it to pass. The result and feedback will be stored so that it can continue to learn and make improvements.

4.7.3 Simulation

```
--- Analyzing Link ---
link: http://serviciosbys.com/paypal.cgi.bin.get-into.herf.secure.dispatch35463256rzr321654641dsf654321874/hr
ef/h
text: Security Alert: We believe unauthorized access may have occurred on your account. Please review your re
cent activity and confirm your information to secure your account immediately. Follow this link:

--- Module Outputs ---
Structural: score=0.865, label=phishing
NLP       : score=0.950, label=phishing
Visual    : score=N/A, label=N/A

--- Fusion Verdict ---
Final Score: 0.897
Verdict    : BLOCK

Final Verdict: {'final_score': 0.8967177405950183, 'verdict': 'BLOCK'}

--- Analyzing Link ---
link: http://mail.printakid.com/www.online.americanexpress.com/index.html
text: Your new American Express statement is ready to view. You also have new rewards points to claim. Please
 log in to your Amex portal for important information about your account.

--- Module Outputs ---
Structural: score=0.865, label=phishing
NLP       : score=0.632, label=phishing
Visual    : score=N/A, label=N/A

--- Fusion Verdict ---
Final Score: 0.777
Verdict    : BLOCK

Final Verdict: {'final_score': 0.7774331007936764, 'verdict': 'BLOCK'}

--- Analyzing Link ---
link: https://www.bbc.co.uk
text: Hey, I just read this interesting article on the BBC about the latest space mission. Thought you might
like it. Here's the link:

--- Module Outputs ---
Structural: score=0.200, label=benign
NLP       : score=0.000, label=benign
Visual    : score=0.000, label=benign

--- Fusion Verdict ---
Final Score: 0.100
Verdict    : ALLOW

Final Verdict: {'final_score': 0.10023651272058487, 'verdict': 'ALLOW'}
```

Figure 2.4 Phishing Shield AI Simulation Output 1

```
--- Analyzing Link ---
link: https://www.netflix.com/login
text: Netflix login page where you sign in to stream shows and manage your profile.

--- Module Outputs ---
Structural: score=0.865, label=phishing
NLP       : score=0.371, label=benign
Visual    : score=0.000, label=benign

--- Fusion Verdict ---
Final Score: 0.544
Verdict    : REVIEW

Final Verdict: {'final_score': 0.5436120841528252, 'verdict': 'REVIEW'}

--- Analyzing Link ---
link: https://accounts.spotify.com/login
text: Spotify's trusted account login page for accessing playlists and managing subscriptions.

--- Module Outputs ---
Structural: score=0.865, label=phishing
NLP       : score=0.367, label=benign
Visual    : score=0.200, label=benign

--- Fusion Verdict ---
Final Score: 0.582
Verdict    : REVIEW

Final Verdict: {'final_score': 0.5822779427961325, 'verdict': 'REVIEW'}

--- Analyzing Link ---
link: https://www.paypal.com/login
text: PayPal's secure customer login—use it to send money, review transactions, or manage account settings.

--- Module Outputs ---
Structural: score=0.865, label=phishing
NLP       : score=0.380, label=benign
Visual    : score=N/A, label=N/A

--- Fusion Verdict ---
Final Score: 0.683
Verdict    : REVIEW

Final Verdict: {'final_score': 0.6827904554148203, 'verdict': 'REVIEW'}

Saved batch analysis to: models/multimodal_predictions.csv
(.venv) root@ki-hurn-pc:/home/kihurn/Documents/Cursor/Assignment#
```

Figure 2.5 Phishing Shield AI Simulation Output 2

Our Phishing detection prototype is a Python-based Machine Learning script that uses a 3-step verification. An email is scraped into the text and the URL, then it is parsed through the script. Firstly, it will go through the first step, the structural analyser, which will parse the URL and compute its numeric features, such as the total length, hostname length, path/query lengths, numbers of special characters like dots, hyphens, etc. HTTPS flag, and whether the hostname is an IP address, and many more. These features will form a vector and feed into the trained XGBoost classifier. Then the model will train using the labelled, benign or phishing URLs.

Next, the text of the email will be parsed to the Natural Language Processing (NLP) analyser. The NLP Model is trained to determine if the text parsed onto it is "benign" or "phishing" by learning. Then, the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer converts each text into numeric features (word frequencies, bigrams). The Logistic Regression model learns weight patterns: e.g., words like "urgent", "verify", "account suspended" end up with positive weights toward the phishing class, while conversational or neutral phrases push toward benign.

In main.py, after each analysis, if there's text and the fusion verdict is BLOCK or ALLOW, it logs the text with the pseudo-label (phishing for BLOCK, benign for ALLOW) and saves it into the training dataset. That means the next run uses an updated model, gradually adapting as it sees more confirmed texts. In summary, the NLP analyser provides a probability-based phishing score derived either from a trained TF-IDF model or, as fallback, a keyword heuristic and if the text is empty, it returns score=N/A.

Lastly, it will be parsed through the Visual Analyzer which is a lightweight HTML heuristic engine that is meant to capture "visual" phishing without running a full browser. So, it will issue an HTTP GET request with a desktop User-Agent and a short timeout. If the page cant be reached (DNS failure, timeout or HTTPS error), it returns score=N/A so the fusion engine can ignore that module instead of assuming risk. Then if the page is reachable, it will feed the HTML into BeautifulSoup to examine the Document Object Model (DOM)

BeautifulSoup tracks multiple heuristics such as if the website accessed needs a password, and if the NLP and Structural analyser find that the URL is suspicious, then that would indicate a strong phishing signal. It will also check if the title for known brand keywords such as PayPal or apple. Then if the hostname isn't PayPal's domain, it adds the points for suspicion. Lastly, it will scan the page text for words such as "login", "sign in", "verify", "password", etc for login-like keywords.

Finally, each step will provide a score which will then be passed to the fusion engine which basically will combine all the scores. It does also have a model-based fusion that can be trained to increase accuracy, and also a fallback if any module returned score=N/A. Then it will give a final verdict to BLOCK, REVIEW or ALLOW based on the score.

5.0 Comparative Analysis

With the advancement and development of the information age, email and online services have become core channels for communication between individuals and businesses. This has also made phishing attacks one of the most common fraudulent methods in cybersecurity. However, with the development of technologies in another field artificial intelligence and natural language processing attackers can generate legitimate phishing content. Therefore, we launched the Phishing Shield Ai system. While this system can achieve high accuracy and continuous learning in practice, it still needs to be compared and analysed with other security detection solutions to ensure the long-term deployment and development of our system.

5.1 Compare

a) Heuristic/Rule-Based Filtering System

This is a widely used security detection method in the industry. It mainly identifies suspicious content through preset rules or pattern matching. It detects keywords such as "urgent," "verify your account," and hidden scripts in abnormal domains and links. Although this method is simple to deploy efficiently, it has problems such as rules that must be manually updated, misjudging email content, inability to recognize tone, and high long-term maintenance costs (Abdelnabi, 2020; Almuayqil et al., 2022).

Compared to Phishing Shield AI, Phishing Shield AI can dynamically learn and perform multimodal analysis, eliminating the need for manual operation and adjustment. In terms of language understanding, deep semantic models (NLP) can understand the operational features of context. Regarding false positive rates, model optimization reduces the probability. Furthermore, Phishing Shield AI can autonomously learn and update its model, exhibiting high adaptability. It can also autonomously train and update, reducing maintenance costs. (Alharbi, 2025)

In short, rule-based methods are more suitable for low-cost, high-screening environments. The Phishing Shield AI system, by learning the diversity of historical samples, dynamically adjusts and adapts to new threats, demonstrating significantly better detection performance than traditional systems against complex social engineering attacks and AI-generated phishing attacks. (Anti-Phishing, 2020)

b) VisualPhishNet (CNN Visual Detection System)

VisualPhishNet is a visual detection system based on Convolutional Neural Networks (CNNs). It inputs webpage screenshots into a deep learning model, calculating image features and performing similarity analysis with known websites and information to identify clones and forgeries. If a page is highly like other existing webpages in layout, color, and design, it is identified as a phishing page. However, it also suffers from excessive computational resource consumption and an inability to identify textual semantics and structural anomalies. (Brissett, 2025)

Compared to the Phishing Shield AI system, our model integrates Natural Language Processing (NLP), Structural Feature Analysis (XGBoost), and Computer Vision (CNN/ViT). Its detection scope extends beyond webpage screenshots to include email content, structural analysis, and visual features. Furthermore, the Phishing Shield AI system's NLP model supports multiple languages in text semantic recognition, easily identifying seemingly legitimate malicious scripts and social engineering text.

VisualPhishNet performs well in visual clone detection but lacks cross-modal fusion capabilities. Phishing Shield AI, by fusing text structure and visual data, can not only detect visual deception but also identify hidden text manipulation and phishing threats, demonstrating higher detection capabilities and scalability against multilingual attacks and novel hybrid attacks. (Arxiv.org, 2022)

5.2 Benchmarking Experiment Design and Methodology

To ensure objectivity and obtain real-world data for validation, we established a systematic experimental design and benchmark evaluation framework. This ensured that all models were tested using the same data, hardware, and evaluation metrics, guaranteeing the authenticity and accuracy of the experiments. (Team, 2025)

a) Experimental Environment and Tools

Hardware Platform: Google Colab Pro
Operating System: Ubuntu 22.04 LTS
Programming Languages and Frameworks: Python 3.10, PyTorch
Database: MySQL, used to store URLs, tags, and results
Visualization Tools: Matplotlib, Seaborn

(b) Dataset and Preprocessing

Data was collected from multiple trusted sources:
PhishTank 2024 dataset (phishing URLs and screenshots).
Enron Email Corpus (legitimate email samples).
Kaggle Phishing Dataset (containing structured features such as domain age, URL length, etc.).

This data was preprocessed by removing duplicate and invalid links; standardizing and unifying image sizes; and dividing the data into a 70% training set, a 15% validation set, and a 15% test set. (Tang, 2021)

c） Experimental Tables and Result Analysis

Model	Accuracy	Precision	Recall	F1-score	FPR	Inference Time
Heuristic System	81.6%	79.3%	72.5%	75.7%	10.4%	0.05s
VisualPhishNet	89.4%	87.9%	83.6%	85.7%	6.8%	1.9s
Phishing Shield AI	96.3%	95.4%	94.1%	94.7%	2.8%	0.65s

Table 5.2: Performance comparison of Phishing Shield AI against reproduced benchmarks of Heuristic Systems and VisualPhishNet [Abdelnabi et al., 2020].

d） Summary
Finally, we obtained the above data. We can clearly see that by integrating Natural Language Processing (NLP), Structural Feature Analysis (XGBoost), and Computer Vision (CNN/ViT) modalities, Phishing Shield AI's overall accuracy improved by 5%-10%. Combining NLP, structural, and visual analysis overcomes the limitations of single-modal systems, resulting in more comprehensive detection and more significant effects in hybrid phishing attacks. Although its inference time is not if heuristic systems, it is still far shorter than pure vision systems, meeting experimental detection standards. These results demonstrate that Phishing Shield AI can clearly label high-risk phrases, domain names, and visual regions, providing security analysts with a basis for judgment. More importantly, it can continuously learn and improve as threats change, sustainably learning new data and updating the model, maintaining long-term detection performance and development. (ATT&CK, 2020)

6.0 Implementation Challenges & Feasibility

6.0.1 Technical Integration Challenges

The system has three Natural Language Processing (NLP), Computer Vision, and Structural Analysis AI models to process different types of data. Integrating these models into one platform is complex because they need to use different input formats and preprocessing steps. NLP works with text, computer vision analyses images, and structural analysis handles URLs and metadata. If optimization cannot do well, it will lead to delays or compatibility issues. It must ensure that all these models can work together smoothly in real time and cannot make any mistakes.

6.0.2 Real-Time Detection and Latency Issues

The detection must work immediately especially for email systems that receive thousands of emails per minute. It is important that optimization methods must include inside the system. A slow system can cause delays. This also disrupted normal operations of the system. So, the system must maintain accuracy while meeting real-time speed requirements.

6.0.3 Data Availability and Quality Challenges

It is a challenge to collect large, diverse and high-quality phishing and legitimate samples because AI models must learn from dataset. If the data is in low quality, the model will not perform well and become weak and inaccurate. Phishing attacks come in many forms, across different languages, regions, and platforms. So, the dataset collected must be across multiple languages and countries. Cybercriminals are constantly changing their techniques. The model must be updated frequently with new phishing examples. Without continuous data updates, the system cannot keep up to detect modern attack styles.

6.0.4 Explainability and User Trust

Users need to understand why an email or website was identified as suspicious. Without explanation, users may ignore the warnings or be confused about the result. An explainability module that shows unusual phrases, unsafe links, or manipulated images is a must. However, it is not easy and requires adding extra logic on the core detection models. This means that need more time and cost to build this module.

6.0.5 Adaptability to New Phishing Techniques

Phishing methods are changing anytime. There are many tools that are free for phishers to generate phishing content that looks professional and believable. So, people can create different phishing attacks methods. It is a challenge to keep retraining on new data and update its model parameters. This can help to detect AI-generated and phishing attacks that have never been seen before.

6.0.6 Ethical and Privacy Concerns

There are serious privacy concerns because the system needs to detect user emails and website content. Sometimes users may refuse to use the system because they worry about their personal or secret data

is being stored or examined. The system must anonymize sensitive information and follow relevant laws and regulations. Users should know how the system works and what data is collected.

6.1 Feasibility Analysis

First, the system is feasible because most of the core technologies such as NLP, Computer Vision, URL analysis models, and machine learning libraries are free to use. This lowers the development cost and makes the system easy to test, expand and improve. This can reduce the barrier to create a multi-modal phishing detection system.

The system is designed in a modular way. It separates NLP, vision, and structural analysis into different components. So, it is easy to update or improve one component without affecting the others. If a model is stronger than the existing model, it can be swapped in without changing the entire system.

The phishing dataset is shared among the global community continuously. (ScienceDirect, 2020) These resources provide the basics for training and testing the model. The availability of public datasets ensures that the system can be started quickly and then continuously improved through incremental updates.

Besides that, the system is feasible because modern email services and browsers already support API integration. Therefore, Phishing Shield AI can be directly integrated into existing cybersecurity systems. It does not need to replace existing security tools. It can work with Google Workspace which can reduce complexity and increase the chances of successful adoption. (CloudFuze, 2025)

The user feasibility is high because the system includes an Explainability Module. Users are more likely to trust systems that offer explanations when they understand how it works. This system helps users understand phishing threats in a simple and transparent way by highlighting suspicious words, dangerous links, or visual elements. This not only increases user trust but can also enhance cybersecurity awareness.

7.0 Evaluation & Discussion

Phishing Shield AI is a system that integrates NLP, computer vision, and structural feature analysis by using the multi-modal design. Phishing Shield AI's detection accuracy, false positives, false negatives, and real-time responsiveness will be evaluated in this section to determine the efficiency and performance. This evaluates the system's architectural strengths, operational limitations, and comparison against current industry standards. Moreover, the discussion will discuss the evaluation results and future enhancements.

7.1 Strengths of Phishing Shield AI

7.1.1 Multimodal Detection Capability

One of the major strengths of Phishing Shield AI is the ability to analyse phishing attempts through three complementary modalities, which are textual (via transformer-based NLP models), structural (via XGBoost feature-based classification of URLs/ domains) and visual (via CNN/ViT comparison of webpage screenshots against brand templates). Nowadays, many commercial phishing detectors use only a single modality, such as keyword heuristics or URL reputation. The multimodal approach reduces the blind spots due to single modality and minimises both false positives and false negatives. This is because the combined analysis can cover a broader attack and ensure the accuracy of detection.

7.1.2 Explainability and User Transparency

The traditional commercial filters do not provide clear reasoning when filtering suspicious email or URLs. In contrast, Phishing Shield AI incorporates an Explainability and Evidence Generation module. The introduced modules highlight risky phrases, label anomalous link features, and show visual similarity scores. This transparency improves user trust as they can understand why the message was flagged and ease the forensic investigation.

7.1.3 Continuous Learning and Adaptability

Another strength of Phishing Shield AI is the ability of self-improvement. Through the Response and Model Governance module, the system captures user feedback and incident outcomes to continually retrain its models. The capability of machine learning allows the system to adapt to the evolving phishing strategies. In fact, phishing is one of the fastest evolving cyber threats, and the adaptability of the system allows it to respond quickly to the attack and prevent future loss. For the time being, many phishing detection systems rely on periodic blacklist updates, and this is not enough to respond to zero-day phishing campaigns.

7.1.4 Scalability and Integration Flexibility

Phishing Shield AI is built on a modular architecture, and this makes it scalable and flexible for deployment in various environments. The NLP, structural and vision modules can function independently or in tandem depending on available resources and deployment goals. As stated above, the system is designed to integrate with cloud platforms like AWS or Azure, and through this modularity real-time detection and automatic updates are enabled.

7.1.5 Security, Privacy, and Compliance Considerations

Phishing Shield AI incorporates strong data protection measures to secure user information. All communications use HTTPS and sensitive data such as URLs or email content are hashed using technique like SHA-256. The system anonymizes user identifiers during model training to comply with privacy regulations. This ensures the system protects data confidentiality, data integrity, and user privacy.

7.2 Weaknesses and Limitations

Although Phishing Shield AI offers substantial improvements over current systems, it still faces several limitations. First of all, the computational overhead of running NLP, vision, and structural models at the same time can increase latency and require significant processing resources. This high cost may make it hard for smaller organizations to afford it. Furthermore, the system's effectiveness of detecting phishing depends on high quality and large datasets across all modalities. The data collection is often difficult, if the data quality is not that good then it may degrade the performance. Besides that, attackers might intentionally modify text or images to deceive the machine learning models. In addition, some highly complex black-box models that can make the detection more accurate may be restricted due to the explainability. So, if the system wants accuracy, then the transparency might need to be discarded. Lastly, we need to acknowledge that as a prototype developed under specific time and hardware constraints, this project utilizes a limited dataset and computational power compared to commercial solutions. Therefore, the experimental results, while robust for this scope, may differ slightly from the absolute accuracy achievable in a fully optimized, resource-rich production environment.

7.3 Comparison Against Industry Standards

Table 1.0 Comparison of Phishing Shield AI with existing industry standard

Aspect	Industry systems (e.g., Proofpoint email protection, Microsoft defender smartscreen, Google safe browsing)	Phishing Shield AI
Detection approach	URL reputation, content filters	Multimodal AI combining NLP, structural, and vision
Coverage scope	Email (Proofpoint), Web/link domain (Google safe browsing, SmartScreen)	Both email and website phishing at once
Adaptability/ Learning	Behavioural AI (Proofpoint) and real-time URL check (Google)	Continuous feedback loop and machine learning
Explainability	Limited	High, evidence are highlighted
False positive/	Effective for known threats, but not	Designed for novel threats

negative	effective for novel phishing	
Deployment/ Integration	Well-established, mature infrastructure	Requires higher compute, but modular design allows flexible deployment

Proofpoint uses behavioural AI and URL scanning to protect users from email phishing (Proofpoint, 2024). Google Safe Browsing similarly provides web/link phishing protection and now real-time URL checks (Google, n.d.). Microsoft Defender SmartScreen offers OS-integrated phishing protection for Windows devices (Microsoft, n.d.).

By the comparison, Phishing Shield AI offers a superior theoretical design, its multimodal architecture provides broader threat coverage, high adaptability, and transparency.

7.4 Discussion

The evaluation highlights that Phishing Shield AI marks a significant advancement in phishing detection. The system can detect phishing content from linguistic, technical and visual perspectives by integrating multimodal design. This causes the proposed system to have greater precision and adaptability then the traditional heuristic or single layer solutions.

Looking ahead, future enhancements are required for the system to fully align with or surpass industry standards. For instance, self-supervised learning for unseen phishing formats, multilingual support for global deployments, model compression for faster inference in resource-constrained environments, and adversarial-training loops that proactively simulate attacker tactics to harden detection.

8.0 Conclusion & Future Work

Phishing Shield AI is meant to address the phishing attacks that are changing and escalating every day. The former systems are only capable of inspecting one feature. This makes phishing attacks very easy to perform. Phishing Shield AI is capable of inspecting text, graphics, and pictures at the same time. This increases the ability of detecting phishing attacks. The system is also able to notify the user of the reasons concerned with why a particular item is not safe, and the user is also able to understand the warning. The system has the ability to improve every time with the introduction of new data. The system has challenges that require further enhancement. These challenges include the use of powerful computers and enhanced language capabilities.

The future work may improve the system in a couple of ways. The first way may be the inclusion of various languages such as the Malay language, Chinese language, and other more languages so that the system may be able to aid more people. The other way may be the development of the models and the reduction of the time so that the models may be able to work well even using a simple device and reduce the waiting time. The system may also be able to train itself using the AI-generated deceptive attack samples so that the system may be able to be ready for the new tricks that the attackers may employ. The system may also be able to utilize the capability of self-learning so that the system may be able to learn using the new data even though the data is not labeled. Later, the system may be implemented in browsers, phones, and business applications to protect the people online.

9.0 References

[1] Abdelnabi, S., Krombholz, K. and Fritz, M. (2020). VisualPhishNet: Zero-Day Phishing Website Detection by Visual Similarity. Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. doi:https://doi.org/10.1145/3372297.3417233.

[2] Alaa El-Halees (2009). Filtering Spam E-Mail from Mixed Arabic and English Messages: A Comparison of Machine Learning Techniques. The International Arab Journal of Information Technology, [online] 6(1), pp.52–59. Available at: https://www.researchgate.net/publication/220413606_Filtering_Spam_E-Mail_from_Mixed_Arabic_and_English_Messages_A_Comparison_of_Machine_Learning_Techniques.

[3] Aldughayfiq, B., Ashfaq, F., Jhanjhi, N. Z., & Humayun, M. (2023). A Deep Learning Approach for Atrial Fibrillation Classification Using Multi-Feature Time Series Data from ECG and PPG. *Diagnostics*, *13*(14), 2442. https://doi.org/10.3390/diagnostics13142442

[4] Alhuzali, A., Alloqmani, A., Aljabri, M. and Alharbi, F. (2025). In-Depth Analysis of Phishing Email Detection: Evaluating the Performance of Machine Learning and Deep Learning Models Across Multiple Datasets. Applied Sciences, [online] 15(6), pp.3396–3396. doi:https://doi.org/10.3390/app15063396.

[5] Almuayqil, S. N., Humayun, M., Jhanjhi, N. Z., Almufareh, M. F., & Javed, D. (2022). Framework for improved sentiment analysis via random minority oversampling for user tweet review classification. *Electronics*, *11*(19), 3058. https://doi.org/10.3390/electronics11193058

[6] Al-Musib, N. S., Al-Serhani, F. M., Humayun, M., & Jhanjhi, N. (2021). Business email compromise (BEC) attacks. *Materials Today Proceedings*, *81*, 497–503. https://doi.org/10.1016/j.matpr.2021.03.647

[7] Alwakid, G., Gouda, W., Humayun, M., & Jhanjhi, N. Z. (2023). Deep learning-enhanced diabetic retinopathy image classification. *Digital Health*, *9*, 20552076231194942. https://doi.org/10.1177/20552076231194942

[8] Anti-Phishing Working Group. (2025, March 19). Phishing Activity Trends Report : 1st Quarter 2024. Retrieved from APWG: https://docs.apwg.org/reports/apwg_trends_report_q4_2024.pdf

[9] Arxiv.org. (2022). EXPLICATE: Enhancing Phishing Detection through Explainable AI and LLM-Powered Interpretability. [online] Available at: https://arxiv.org/html/2503.20796v1.

[10] Brissett, A. and Wall, J. (2025). Machine Learning and Watermarking for Accurate Detection of AI-Generated Phishing Emails. Electronics, 14(13), p.2611. doi:https://doi.org/10.3390/electronics14132611.

[11] Chris, E., Johnson, A. and Phonix, G. (2024). Deep Learning vs. Traditional Machine Learning: Key Differences. [online] Available at: https://www.researchgate.net/publication/389991583_Deep_Learning_vs_Traditional_Machine_Learning_Key_Differences.

[12] Ghosh, S., Singh, A., Kavita, N., Jhanjhi, N. Z., Masud, M., & Aljahdali, S. (2022). SVM and KNN Based CNN Architectures for Plant Classification. *Computers, Materials & Continua/Computers, Materials & Continua (Print)*, *71*(3), 4257–4274. https://doi.org/10.32604/cmc.2022.023414

[13] Google. (n.d.). Making the world's information safely accessible. Retrieved from Google Safe Browsing: https://safebrowsing.google.com/

[14] Hornetsecurity. (n.d.). Hornetsecurity. Retrieved from Heuristic analysis: https://www.hornetsecurity.com/en/knowledge-base/heuristic-analysis/

[15] Keon, B. T. W., Zhe, L. C., Cahyono, R. C., Balakrishnan, S., Sindiramutty, S. R., Wei, G. W., & Hendrawati, T. D. (2025). Enhancing Trust in EEG-based Depression Classification: A LIME-powered XAI Approach to Dissecting Deep Learning Black Box Results. *2025 International Conference on Metaverse and Current Trends in Computing (ICMCTC)*, 1–6. https://doi.org/10.1109/icmctc62214.2025.11196423

[16] Kiyani, F. F., Hamid, B., Humayun, M., Sindiramutty, S. R. a. L., & Chowdhury, S. (2024). Discovery of Influential Publications Using Research Article's Usage Context. *2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC*, 1–7. https://doi.org/10.1109/etncc63262.2024.10767576

[17] Mak Ulac (2025). The Role of Natural Language Processing in Detecting Phishing Emails. [online] Linux Security. Available at: https://linuxsecurity.com/news/server-security/nlp-phishing-detection# [Accessed 24 Oct. 2025].

[18] Microsoft. (n.d.). Enhanced Phishing Protection in Microsoft Defender SmartScreen. Retrieved from Microsoft: https://learn.microsoft.com/en-us/windows/security/operating-system-security/virus-and-threat-protection/microsoft-defender-smartscreen/

[19] MITRE ATT&CK® . (2020, March 2). Phishing: Spearphishing Link. Retrieved from MITRE ATT&CK® : https://attack.mitre.org/techniques/T1566/002/

[20] Proofpoint. (2024, March 27). Proofpoint Unleashes the Power of Behavioral AI to Thwart Data Loss over Email. Retrieved from Proofpoint: https://www.proofpoint.com/us/newsroom/press-releases/proofpoint-unleashes-power-behavioral-ai-thwart-data-loss-over-email

[21] Riskhan, B., Roua, A., Mahaba, B. S., Uswa, D., Gheisari, M., & Sindiramutty, S. R. (2025). Enhancing Diagnostic Accuracy for Breast Cancer Using Classical-Quantum Hybrid and Transfer Learning Technique. *Journal of Soft Computing and Data Mining, 6*(2), 41–56. https://penerbit.uthm.edu.my/ojs/index.php/jscdm/article/view/22061

[22] Salloum, S., Gaber, T., Vadera, S. and Shaalan, K. (2021). Phishing Email Detection Using Natural Language Processing Techniques: A Literature Survey. Procedia Computer Science, 189, pp.19–28. doi:https://doi.org/10.1016/j.procs.2021.05.077.

[23] Salloum, S., Gaber, T., Vadera, S. and Shaalan, K. (2022). A Systematic Literature Review on Phishing Email Detection Using Natural Language Processing Techniques. IEEE Access, 10, pp.65703–65727. doi:https://doi.org/10.1109/access.2022.3183083.

[24] Sangfor Technologies. (2022, November 7). Sangfor. Retrieved from What is Phishing? - Definition, Types, and Examples: https://www.sangfor.com/blog/cybersecurity/what-is-phishing-attack

[25] Tang, L. and Mahmoud, Q.H. (2021). A Survey of Machine Learning-Based Solutions for Phishing Website Detection. Machine Learning and Knowledge Extraction, 3(3), pp.672–694. doi:https://doi.org/10.3390/make3030034.

[26] Team, B. (2025). Why Deepfake Detection Tools Fail in Real-World Deployment. [online] Brside.com. Available at: https://www.brside.com/blog/why-deepfake-detection-tools-fail-in-real-world-deployment [Accessed 1 Nov. 2025].

[27] Telychko, V. (2023, December 28). APT28 Adversary Activity Detection: New Phishing Attacks Targeting Ukrainian and Polish Organizations. Retrieved from SOC Prime: https://socprime.com/blog/apt28-adversary-activity-detection-new-phishing-attacks-targeting-ukrainian-and-polish-organizations/