Naif Arab University for Security Sciences
Journal of Information Security and Cybercrimes Research
مجلة بحوث أمن المعلومات والجرائم السيبرانية
https://journals.nauss.edu.sa/index.php/JISCR

XAI for Improving the Security of ML-Based Spoofing Email Detectors

Niddal Hassan Imam

Saudi Electronic University, Kingdom of Saudi Arabia.

Received 12 July 2025; Accepted 5 Oct. 2025; Available Online 31 Dec. 2025

Abstract

Email is an essential part of our daily communication as it is one of the primary communication methods. Cyber-attacks against email systems and their users have been increasing over the years for different reasons. For example, Phishing is among the most common attacks that target email users with the intention to induce them to disclose personal information. Typically, attackers use email spoofing attacks as the initial step in launching a phishing attack. Most existing studies focus on phishing attacks, overlooking email spoofing attacks. Several mitigation methods have been proposed to defend against email system-related attacks using Artificial Intelligence (AI) and Machine Learning (ML) models. However, the literature has shown that these automated models are vulnerable to adversarial examples that can cause them to produce inaccurate predictions. The objective of this study is to identify evolving trends in email spoofing mitigation methods that uses ML and to highlight limitations and gaps. The review distinguishes itself by carefully reviewing the latest articles published between 2020 and 2024, stating their pros and cons. The results indicate a lack of studies focusing on email spoofing attacks, which is a crucial step in phishing attacks. Additionally, it reveals that most existing studies fail to consider the design of adversary-aware ML-based detectors for spoofed emails. Thus, an adversary-aware framework for detecting spoofed emails was proposed, and multiple experiments were performed to simulate possible adversarial attackst.

I. Introduction

The advancement in data communication technologies has significantly changed users' experiences, both positively and negatively. While people and businesses rely heavily on email, SMS, and social media for communication, cyberattacks have introduced significant technical, financial, and social threats [1]. Cyber attacks have become a serious threat not only to the communication system but also aiming to steal users' money and identity [2]. They have been used as tools to compromise users' privacy by identity theft, fraud, or social engineering techniques [3]. As emails have become an essential part of our daily lives and are considered the most widely used form of data communication over the Internet, attackers are using them to launch cyberattacks, such as spoofing, phishing, and Business Email Compromise (BEC) attacks [4]. Attackers have targeted email users, as they tend to trust messages received via email more than those received through other communication systems.

Keywords: Adversarial examples, email spoofing, explainable artificial intelligence (XAI), label flipping attack, machine learning

Production and hosting by NAUSS

* Corresponding Author: Niddal Hassan Imam
Email: n.imam@seu.edu.sa

doi: 10.26735/INJZ6869

1658-7782© 2025. JISCR. This is an open access article, distributed under the terms of the Creative Commons, Attribution-NonCommercial License.

Emails are typically used for formal communication, unlike social media platforms. Users communicate with their bank and internet providers via email; employees exchange emails with colleagues to perform their job duties. The frequent use of email and the value of information it exchanges attracted attackers [5].

Phishing involves the use of fraudulent messages to deceive recipients into revealing sensitive information through various communication systems, including email, URLs, SMS, social media, and online games [6], [7]. Phishing is commonly carried out through email spoofing or texting, and it involves tricking the user into entering personal information on a fake website [4]. The most widely used phishing attack is email phishing [8], and email spoofing is a crucial step in launching such an attack. Email spoofing is a cyberattack in which an attacker creates a forged message by manipulating the sender's email address or content, making it appear to the victim as if it has been sent from a genuine sender [4]. Typically, spoofed emails aim to expose victims' personally identifiable information (PII), which can be exploited for identity theft [9]. It is one of the most common forgery techniques that does not require knowledge or effort from adversaries. Thus, email spoofing can be used as a reliable indicator of phishing attacks [10].

Artificial intelligence (AI) has been widely utilised to enhance email security by providing rapid and accurate predictions [11]. Several machine Learning (ML) algorithms have been used for developing spoofed email detectors that extract email features, such as header [10], [12], [13], [1], [4] or content [14], [15], [2], [16], [3], [11], [17], [18], [19], [9]) and then classifying emails based on the learned behaviors. Traditional ML algorithms, such as Naive Bayes, Random Forest, and logistic regression, have been used for building detectors that analyse emails' header fields, including FROM, TO, or DATE. Additionally, advanced deep learning models have been recently employed in spoofed email detectors. For detecting spoofed emails by analyzing their body, Natural Language Processing (NLP) techniques that can extract textual features from emails have also been used [19].

The adoption of ML algorithms for detecting spoofed emails has shown lots of success in terms of detection accuracy; however, it raises concerns about the vulnerability of these automated solutions if they were not developed for adversarial settings. Designing an ML-based model to solve a cybersecurity issue necessitates consideration of the model's security. The robustness of ML-based models against adversarial examples (i.e., inputs carefully crafted to manipulate ML models) has become subject to increased interest in the research community [20]. Researchers have been studying the security of ML since 2004, when a group of researchers Dalvi et al., [21] developed a framework and algorithms to detect adversarial activities. Designing proactive spoofed email detectors rather than traditional reactive ones is crucial, as reacting to detected attacks will never prevent future attacks. Designing a detector that can anticipate adversaries' attacks proactively enables designers to develop suitable defence methods before an attack occurs [22].

Recent works in the field of Adversarial ML (AML) have shown the importance of designing adversary-aware ML-based models that are robust, adaptable and explainable. As detection models improve, adversaries' attack methods evolve accordingly. This arms race has created a newly emerging type of adversaries that target these automated cybersecurity solutions by attempting to evade detection or degrade the performance of detectors [23]. Susceptibility of ML models are also susceptible to adversarial drift, which can result from a distributed denial-of-service (DDoS) attacks [24]. Understanding ML's decisions, predictions, and performance is critical not only for users but for system designers, as it helps them effectively manage the systems [25]. Explainable AI (XAI) is widely used nowadays to provide reasoning behind AI's predictions. The explainability of ML-based models enables designers or developers to investigate if a model is under adversarial attacks and debug it if needed. Various libraries for ML model explainability/ interpretability exist; three popular ones are SHAP [26], LIME [27], and ELI5 [28]. SHAP (SHapley Additive exPlanations) is a tool for determining the contribution of each feature to the model's prediction. Additionally, LIME (Local Interpretable Model-agnostic Explanations) was employed to interpret the model's prediction for a single instance. ELI5 stands for "explain like I am 5.", and it aims to explain the prediction of any model [29].

Although a large number of studies have analysed phishing attacks, more attention needs to be given to email spoofing, which is an essential step in the life-cycle of phishing attacks. Email spoofing is not a type of phishing attack; it is a tool that helps an attacker bypass deployed detectors [10]. Mitigating email spoofing attacks can help reduce the success of phishing attacks, as it enables attackers to bypass detectors. Thus, several existing studies utilise email spoofing as an indicator of phishing attacks [10], [30]. Raising email users' awareness of email spoofing attacks would affect the success of phishing attacks. Moreover, designing adversary-aware ML-based detectors for spoofed emails is now a necessity. Many of the proposed detection methods utilise off-the-shelf ML models, which have recently shown some weaknesses against adversarial examples. Robustness against adversarial examples, the adaptability to adversarial drift and the explainability of ML-based spoofed email detectors need to be considered.

Considering the increasing trends in mitigating email attacks using ML, the objectives of this paper are: (1) to identify the existing ML-based mitigation methods against spoofing email attacks, (2) to identify whether existing ML-based detectors of spoofing email attacks designed for adversarial environment or not (3) to identify the gaps and limitations that exist in the literature (4) to propose adversary-aware ML-based detector and to simulate adversary attacks. Taking into account these objectives, this paper presents a systematic literature review (SLR) that highlights the limitations and gaps in existing ML-based email spoofing mitigation methods in terms of robustness, adaptability, and explainability. Although there are a large number of SLR articles that focus on email phishing detection, to the author's knowledge, this is the first SLR article that focuses on email spoofing detection using ML. To summarize, the main contributions of the research are as follows:

1. It provides a survey of important and relevant research that discusses spoofed email detection using ML.
2. It proposes an adversary-aware framework for detecting spoofed emails.
3. It presents simulations of potential adversarial attacks against spoofed email detectors.

The rest of the paper is structured as follows. Section II summarizes some related works and presents the SLR's results. The results and analysis of scenarios involving potential adversarial attacks against the proposed framework are discussed in Section III. Finally, Section V concludes the paper and discusses future work.

II. Literature Review

A systematic literature review was conducted following the methodology presented by [31], [32], and [33]. This study was conducted in four stages: (1) constructing research questions, (2) defining the search keywords, (3) selecting the list of databases to be used for the search, and (4) defining the inclusion and exclusion criteria. This systematic literature review aims to identify top research findings in the domain of email spoofing mitigation methods that use ML. Current literature was summarised and analysed, and the details of the SLR are presented in this Section

A. Formulating Research Questions.

In this stage, the research questions were formulated by analyzing and identifying the gaps in the existing studies on ML-based email spoofing detectors. The identified limitations, such as the interchangeability of phishing and email spoofing, the design of detectors for both phishing and email spoofing, and overlooking the presence of an adversary that may attack the ML part of email spoofing detectors, in the literature, were considered while constructing the research questions. The primary objective of this study was to determine whether the existing ML-based detectors for email spoofing were designed with security in mind, specifically with regard to robustness against adversarial examples, adaptability to emerging attacks, and explainability for debugging purposes. Table I lists the constructed research questions.

TABLE I
Research Questions

R1	How can ML be utilized to mitigate email spoofing?
R2	Are existing ML-based spoofed email detectors are adversary-aware?
R3	What gaps and open issues emerge from the analysis of the existing state of the art?

B. Defining the Search Keywords

As in [33], the research questions were used to define a list of keywords for the search query. The following is the specified search query:

" email spoofing" AND "ML" OR "DL" OR "AI" OR

"Machine Learning" OR "Deep Learning" OR "Artificial Intelligence".

C. Selecting the Database for search

Related studies use different search engines for systematic review. Google Scholar was selected as the primary search database because it is considered one of the most significant sources of publications [32], [33]. Additionally, Barricelli et Al. [34] suggested using Google Scholar as a database for searching, which helps avoid bias towards any specific publishers.

D. Eligibility Criteria

The inclusion–exclusion criteria were applied at five levels, and ineligible papers were eliminated after each level. A list of inclusion criteria (IC) and exclusion criteria (EC) was defined and applied as follows:

- IC 1: A well-discussed article reports at least three out of the keywords.
- IC 2: Articles written in the English language.
- IC 3: Articles published in a peer-reviewed journal or a conference.
- IC 4: Articles published in the last four years were included (i.e., 2020-2024)
- EC 1: Thesis, news articles, reports, or websites were excluded.
- EC 2: Articles published in a language other than English were excluded.

The initial search query resulted in 1,190 articles

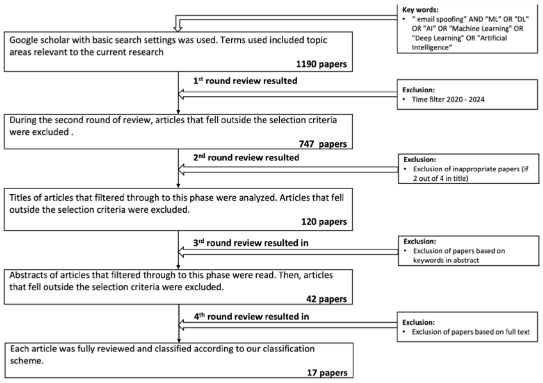
Key words:
• " email spoofing" AND "ML" OR "DL" OR "AI" OR "Machine Learning" OR "Deep Learning" OR "Artificial Intelligence"
Google scholar with basic search settings was used. Terms used included topic areas relevant to the current research
1190 papers
1st round review resulted
Exclusion:
• Time filter 2020 - 2024
During the second round of review, articles that fell outside the selection criteria were excluded .
747 papers
2nd round review resulted
Exclusion:
• Exclusion of inappropriate papers (if 2 out of 4 in title)
Titles of articles that filtered through to this phase were analyzed. Articles that fell outside the selection criteria were excluded.
120 papers
3rd round review resulted in
Exclusion:
• Exclusion of papers based on keywords in abstract
Abstracts of articles that filtered through to this phase were read. Then, articles that fell outside the selection criteria were excluded.
42 papers
4th round review resulted in
Exclusion:
• Exclusion of papers based on full text
Each article was fully reviewed and classified according to our classification scheme.
17 papers

Fig. 1. The diagram depicts the number of Research Articles identified, included and excluded Criteria and the reasons for exclusions

Fig. 2. Word Cloud for the Titles of the Selected Research Papers

Fig. 3. Word cloud for the Abstracts of the Selected Research Papers

from various publishers, including MEDLINE, Web of Science, IEEE Xplore Digital Library, and ACM Digital Library. After applying the time filter for 2020 -2024, the number of articles was reduced to 747. The result of the third level, in which the title of the

TABLE II
Summary of Literature

Title	What	Where	Evaluation	Accuracy	Dataset	Summary
[9]	Stacking algorithm	Contentbased	Accuracy, speed	96	N/A	A new approach for detecting spoofing email was presented. A stacking algorithm, consisting of a base model and a meta model, was employed. The detector is text-based and utilises a classification and clustering model to categorise emails as either phishing or legitimate. After comparing the results of different ML algorithms, they found that the Stacking algorithm outperforms other algorithms in terms of accuracy and speed. Additionally, it has been noted that one of the disadvantages of the stacking algorithm is its complexity.
[4]	RF	header fields	precision, recall, F1-score	98	33080 URLs	An approach for detecting spoofing-based email attacks in an organization through analyzing received and replied emails was proposed. The detector uses seven novel features from URL extraction. It examines the headers of received and replied emails using an ML algorithm after capturing open processes in a browser and identifying the URLs related to open emails. Then, the headers of the live running processes are extracted and analysed by the deployed detection algorithm. Finally, the results are stored in log files. Performance evaluation shows that the model analysis emails faster and produces fewer false positives.
[35]	KNN, LR, RF	Survey	Accuracy	85	Self-collected	An ML-based model for predicting people's risk level of social engineering attacks was proposed. It focuses on different social engineering attacks, including email spoofing. It shows that it is possible to predetermine risk levels of individual in terms of social engineering attacks based on their demographics, technology usage, and personality traits using ML.
[3]	CNN, RCNN	Contentbased	Accuracy	99	Enron, self	It proposes an authorship attribution system that analysis the writing style of an email and predicts if the email is sent or not by a legitimate user. The proposed end-to-end framework uses feature-based and word embedding-based classifiers. Linguistics features (i.e., lexical, structural, syntactical) were converted into numerical values and used by the feature-based classifier. The model is trained to distinguish individual writing styles from others.

Title	What	Where	Evaluation	Accuracy	Dataset	Summary
[12]	SVM, KNN	header fields	Accuracy	90	Self created	It develops an ML-based detector that relies on senders' email structure, such as their personal preferences, email client and infrastructure rather than senders' email content. Three groups of features were extracted from the emails' headers and content. These features make email spoofing significantly more difficult for attackers. SVM and KNN were used to build the detector, and the results show that they can detect spoofed emails with over 90% accuracy. Three adversarial attacks were used to evaluate the robustness, and the results show that if the attackers have access to the victim's email, the detection accuracy decreases to 72%
[10]	KNN, RL	header fields	Accuracy	94	Enron, Uni. Buffalo, SpamAssassin, IWSPA-AP, CSIRO	. It proposed a model for detecting spear phishing that is adaptable to zero-day attacks. It utilised Reinforcement Learning (RL) to select the optimal features that balance accuracy and low feature dimension. Three email spoofing attacks (Adversarial attacks) were launched against the proposed model to evaluate its robustness and adaptability. The results demonstrate that the framework is robust and adaptable, but it has some limitations in terms of runtime and model contextualization. Adversarial attacks using email spoofing were discussed, with a particular focus on spear phishing.
[30]	LR, LDA, SVM	Contentbased, header fields	Accuracy	98	Self-created	It proposed a framework employs sentiment and context-based behavior analysis for the detection of spear-phishing and email spoofing, which is an important tool for detecting spear phishing (spoofed email-based). It utilises a combination of ML and rule-based detectors that enable users to validate malicious emails before they are classified manually. It focuses more on spear phishing attacks.
[36]	NLP, encryption	Contentbased, header field	N/A	N/A	MongoDB Atlas	The research introduces two primary methodologies to combat spoofing: an email filtering system using a machine learning algorithm and an encryption and decryption system using a Caesar Cipher and Python programming language. It detects emails based on approved domains and unapproved domains. A blacklist of domains called MongoDB Atlas was used. The regular Caesar Cipher has been strengthened by the random selection of the shift value every time the program is run. Slightly talks about Adv examples and the importance of HITL. The experiment results have not been provided.

TABLE III
Summary of Adversarial Attacks Coverage In the Literature.

Title	Robustness	Adaptability	Explainability	Novelty
[9]	No	No	No	To compare the performance of a staking algorithm with other ML algorithms in detecting spoofing email based on the textual content
[4]	No	No	No	To propose an ML classifier to identify URLs running in a browser as relevant (for opened emails) or irrelevant (for non-email URLs) instead of capturing the entire physical memory.
[35]	No	No	No	To develop an ML classifier to predict the risk level for an individual of being tricked by social engineering attacks, including spoofed email.
[3]	No	No	No	To propose an authorship verification mechanism that analyses the writing style of a sender and sender-receiver interaction using ML/DL.
[12]	Yes	No	No	To identify emails' features that can distinguish emails without relying on textual content. To design an ML-based detector that is robust against adversarial attacks.
[10]	Yes	Yes	No	To explore the limitations of existing ML-based detectors against unknown or emerging attacks (i.e, Zero-day attacks). To propose a solution that is robust and adaptable against Zero-day attacks.
[30]	No	No	No	To propose a framework that analyses email header, content and attachment to detect spear phishing. A combination of ML and rule-based algorithms was utilised to build the framework.
[36]	Partly	Partly	No	To discusses a possible method of efficiently combating spoofing using NLP-based and ML-based email filtering and an encryption and decryption system using a Caesar Cipher and Python programming language.

articles was read, was 120 articles. Article titles did not include at least 2 out of the 4 keywords were excluded. Then, the abstracts of the remaining articles were screened based on the keywords. They retained 42 articles for full-text review and deemed 17 articles relevant to include in the final full-text extraction. Finally, 11 out of 17 articles were selected for analysis and answering research questions. Figure 1 shows the SLR flow chart.

E. Results and Discussion.

Following the analysis approach in [32], a Word Cloud technique was used to illustrate the close relationship between the selected articles. The bigger and bolder word depicts the frequency and importance. Fig. 2 visualises the word frequencies in the title, and the word cloud for abstracts is shown in Fig. 3 of the selected research papers. The two figures show that the selected string of keywords (i.e., Email spoofing, Machine Learning, and detection) occurs more frequently in the selected articles. However, the term ”phishing” occurs more frequently than ”spoofing,” which suggests a lack of articles that focus on email spoofing.

Table II presents a comparative analysis of existing work on ML-based spoofed email detection techniques. Only 11 out of 17 articles were used in the analysis as they were found to be more focused on the topic. The results show that using ML for detecting spoofed emails by analysing the headers and content has proven to achieve a high accuracy of over 85%. However, existing studies have not considered the robustness, adaptability, and explainability of the ML-based detectors that have been designed. Table III presents the coverage extent of the robustness, adaptability, and explainability of the ML-based detectors designed in the literature. For example, the term “partial” refers to the level of coverage for the selected topics. To the best of the author's knowledge, this paper is the first to explore the three security aspects of email spoofing detection. This suggests that further research in this area is necessary.

Additionally, the conducted systematic literature review aims to address research questions listed in Table I. The results of the SLR will be analysed to answer the research questions in the following subsections.

1) Mitigating Email Spoofing Using ML (RQ1). The identified papers show that ML models have been utilised in three different ways. First, ML can be designed to predict whether an email is spoofed or legitimate by examine the header fields. Authors in [4] designed an ML-based model for detecting spoofing-based email attacks—the designed detector analyses email header extractions (i.e., received and replied). The authors accelerate the detection process by capturing email traces through memory forensics, rather than the entire physical memory. A binary Random Forest (RF) classifier was trained to identify URLs running in a browser as relevant (for opened emails) or irrelevant (for non-email URLs). This helps identify the exact process and only captures their URLs for investigation. In addition, the authors in [12] developed an ML-based detector consisting of a k-nearest neighbours (kNN) classifier, which doesn't require a large training sample, and a multiclass support vector machine (SVM), which is effective in high-dimensional vector spaces. Email header features were grouped into three: behaviour, composition, and transport to characterise the sender of an email. Similarly, authors in [10] built a kNN classifier to predict if an email is spear phishing or legitimate. kNN was chosen because it doesn't rely solely on training data when making predictions. It calculates the distance between data points during the prediction phase. This enables the authors to utilise kNN with a Reinforcement Learning (RL) agent to determine the importance of each feature, which can aid in designing a robust and adaptable detector against Zero-day attacks (i.e., adversarial attacks).

Secondly, the identified papers demonstrate that ML-based detectors can be designed to classify emails by analysing their textual content. Authors in [9] compare the performance of several ML algorithms in detecting spoofed emails. The result shows that a staking algorithm that combines linear regression and logistic regression outperforms other ML algorithms. Authors in [3] proposed an email authorship system for verifying a target's writing style. Different ML/DL algorithms were utilized to build the verification system. The system was modeled as a text binary classification problem to differentiate between the target class (email sent by the declared author) and the non-target class (spoofed email). Two types of features were extracted from email content: feature engineering-based and word embedding-based. CNN and RCNN were utilised for the classification task, achieving an accuracy of 95.3%. In the cited work by [30], the authors propose a multi-layer framework comprising two components: one at the email level and the other at the network level. First, it identifies emails that contain a URL or attachment, and then a three-layer detection engine examines the content, header, and attachment. The first layer uses a rule-based and anomaly-based detector to investigate emails received from external domains (i.e., spoofed emails). In the second layer, LDA (Latent Dirichlet Allocation) is used for sentiment analysis of the emails' subject lines and content. The third layer analysis embedded URLs using Logistic Regression (LR) and attachment using One-class Support Vector Machine (One-class SVM). Authors in [36] proposed an email filtering system that utilises machine learning to distinguish between approved and unapproved email domains. Email domain validation was conceded to detect spoofed emails by using a list of approved vendors stored in a MongoDB Atlas database. An NLP-based model was used for email content classification.

Additionally, ML has been utilized to predict the likelihood that an individual will be a victim of spoofing attacks by evaluating their awareness. Authors in [35] utilise ML to predict the risk level of an individual being tricked by social engineering attacks. Three questionnaires were conducted to create a dataset. Collected data went through data binning, which is ”a process of grouping individual data values into specific bins or groups”. Following this, three feature selection approaches were employed to identify relevant features for the classification task. Finally, the prediction accuracy of six ML algorithms was compared, and kNN, LR, and RF provided the best performance.

F. Adversary-aware ML-based Detectors for Spoofed Emails(RQ2).

The identified papers classify adversarial attacks against ML-based detectors of spoofed email into three categories: Blind spoofing, Known domain, and Known sender. In blind spoofing, it is assumed that adversaries only know the email address of a sender they are trying to impersonate and try to guess the email structure. This attack requires minimal effort and presents a high risk, as it has been shown to bypass several security measures [10]. In a known domain, adversaries have access to one or more emails belonging to senders within the same domain as the spoofed sender. Having access to these emails enables adversaries to craft a spoofed email that mimics some features of the sender they are trying to impersonate. In the final attack, which is called a known sender, adversaries have access to the emails of the sender they are trying to impersonate. This enables them to impersonate the sender accurately.

Authors in [12] evaluate the proposed ML-based model designed to detect spoofed email against two adversarial attacks. The authors built an ML-based detector that classifies emails by considering the sender's profile characteristics. They argue that the proposed detector makes it difficult for adversaries to launch a spoofing attack, as the success depends on how much information about the email structure the adversary can learn. To evaluate the robustness of their detector, three adversarial attacks discussed earlier were employed. The results of the experiment demonstrate that the developed model is capable of detecting spoofed emails when the adversary's knowledge of the email structure of the impersonated sender is limited. Authors in [10] proposed a method to ensure the adaptability of the RL-based detector. They argue that manually extracting features to ensure the model can evolve is problematic. Thus, RL agent was used to generate a new feature subset whenever a new attack type (e.g., zero-day attack) is detected. For example, if the ML-based model was trained to detect blind spoofing attacks, when known domain attacks are encountered, the model will re-train and generate a new feature subset. After conducting experiments, the author concludes that the proposed automated feature extraction is more complex and takes longer to implement than manually engineered features.

Experiments show that the adversary's knowledge level significantly affects attack success.. Although the robustness and the adaptability have been considered in two of the identified papers, the explainability has not been considered.

G. Gaps and Open Issues (RQ3).

Spoofing email attacks, which are considered the initial step for several other attacks, still present a serious threat to email users. Although the identified papers discuss several mitigation methods, some concerns and gaps remain regarding their effectiveness. One such gap identified during the SLR is that most ML-based detectors were not designed for an adversarial environment, where the war between designers and adversaries is never-ending. Only two papers out of 11 evaluate their detectors against adversarial attacks. To overcome this gap, designing adversary-aware ML-based detectors that are robust against adversarial examples, adaptable to emerging attacks, and explainable for debugging needs to be considered. Although robustness and adaptability have been considered, the explainability has not. Helping human decision-making is one of the ultimate goals of using ML. To do so, it should provide a detailed justification for its decisions that facilitates interaction with humans; the explainability of AI/ML plays a crucial role in this regard. Several XAI algorithms have been proposed recently in the literature to provide reasoning behind AI/ML's predictions. Utilising XAI for designing a detector for spoofing email attacks will be considered in this paper.

Furthermore, the effectiveness of existing spoofing email detection depends on the adversary's level of knowledge about the target. If an adversary is capable of exactly mimicking the target's email features (i.e., address, subject, and content), it would be hard for an ML-based detector to distinguish between spoofed emails and legitimate ones [4]. Moreover, the complexity of some models makes them understandable for humans [9] and increases the runtime cost [10]. To overcome these issues, XAI and a Human-In-The-Loop (HITL) mechanism can be integrated with an ML-based detector to aid its predictions. An XAI algorithm can be used to understand the predictions of ML-based detectors, and the HITL approach enables the designer to debug the detector if necessary.

III. Methodology

This section describes the methodology used for developing and evaluating the proposed adversary-aware ML-based detector of spoofing email attacks. It follows methods commonly used for building an ML-based detector in the literature [30], [4], [10], but, in addition, the possible presence of adversaries was considered in each step.

A. Datasets

For our experiments, we used a combination of publicly available Spoofed email and Twitter spam datasets. These datasets were chosen to provide a diverse selection for testing the developed detector. The main Dataset contains 1000 emails with 12 attributes, such as sender, receiver, body, and mismatched sender domain. Each row in the dataset, which provides relevant information about the email, is classified into two classes: 1 if the email is spoofed or 0 if it is not. SPF (Sender Policy Framework): Indicates if the sending IP is authorized to send emails for the domain in the MailFrom address. A ”fail” or ”softfail” is a strong indicator of spoofing. DKIM (DomainKeys Identified Mail): Provides a cryptographic signature to verify the sender’s domain. A ”fail” or missing signature for a known domain is suspicious. DMARC (Domain-based Message Authentication, Reporting & Conformance): Builds on SPF and DKIM to define policy for domains regarding unauthenticated emails. A DMARC ”fail” is a clear sign of spoofing. IP Reputation Score: The reputation of the IP address from which the email originated (lower score for suspicious IPs). The secondary dataset was a Twitter spam dataset . This dataset was chosen as there is a lack of spoofed email datasets. Also, it resembles spoofed email datasets as it contains the messages’ content and header features.

B. Model Selection

Existing studies have proposed different methods to use ML for spoofed email detection, such as Content-based, Header field, and hybrid-based. Although content-based detectors are highly effective against spoofed email attacks [37], attackers can use ChatGPT to generate fake emails that look authentic. Furthermore, the literature demonstrates that designing ML-based models for an adversarial environment without considering the potential adversary that may attack the model is not a realistic approach. Consequently, an existing ML-based model designed for spoofed email detection will be adapted and evaluated considering the robustness, adaptability, and explainability.

The performance of three classic ML algorithms, Random Forest (RF), support vector machine (SVM), and Logistic Regression (LR), which related studies have used, were compared. These three algorithms were used to build an email header detector. The primary dataset (spoofing emails) was used in this experiment. The dataset was split into 70% training and 30% testing datasets. After training the three ML algorithms using the training dataset, they were evaluated using the testing dataset. Based on the results in Table IV, RF was selected for the email header-based detector.

Furthermore, the predictions of three NLP models widely used for text classification tasks—WordLSTM (Word-based Long Short-term Memory), WordCNN (Word-based Convolutional Neural Network), and BERT (Bidirectional Encoder Representations from Transformers) — were evaluated as in [38]. The results in Figure 4 show that BERT achieve the best performance. The spoofed email dataset was used for these experiments, and it was divided into 800 training and 200 testing datasets.

C. Proposed Framework

The architecture of the proposed framework is depicted in environment, where an arms race between system designers and the adversaries is never-ending. It was designed to be robust against adversarial identified examples, adaptable to emerging attacks, and explainable for debugging. It comprises two modules: the detection module and the explanation module. RF and BERT detectors were used for the classification task. Also, SHAP (SHapley Additive exPlanations) was used for determining the contribution of each

TABLE IV
Classification Performance of Three Traditional ML

Models	ROC AUC
RF	0.53
SVM	0.51
LR	0.49

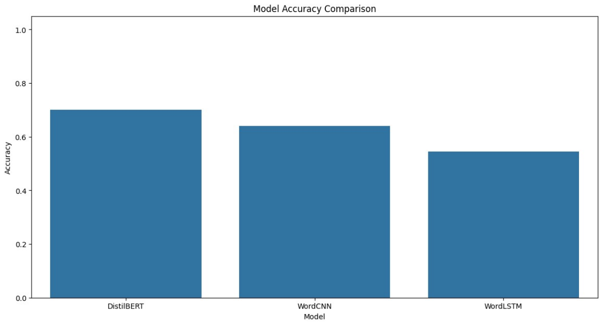

Fig. 4. Classification Performance of Three NLP models

feature to the models' prediction, and LIME (Local Interpretable Model-agnostic Explanations) to interpret the prediction of the models on a single instance. These two XAI models were utilized by the explainability module. This module was used to ensure the adaptability and explainability of the detection model. In the event of disagreement between the detectors, a security analyst is to be alerted to analyze and debug the detector. XAI has been widely used in the literature to provide reasoning behind the predictions of ML-based models. In this study, XAI is utilized to improve adversarial attack awareness of ML-based spoofed email detectors. The main reason for using the XAI to make sure that the developed framework can evolve in the face of emerging attacks. Specifically, the detector was designed to consider adaptability in handling possible adversarial drift that may occur as a result of adversarial activities [39] and to provide explainability to experts (e.g., security analysts).

D. Experimental Settings

This section presents and discusses the experimental results and evaluation. Experiments were performed using the server configuration detailed in Table V. Two models were built and used for implementation and spoofed email detection.

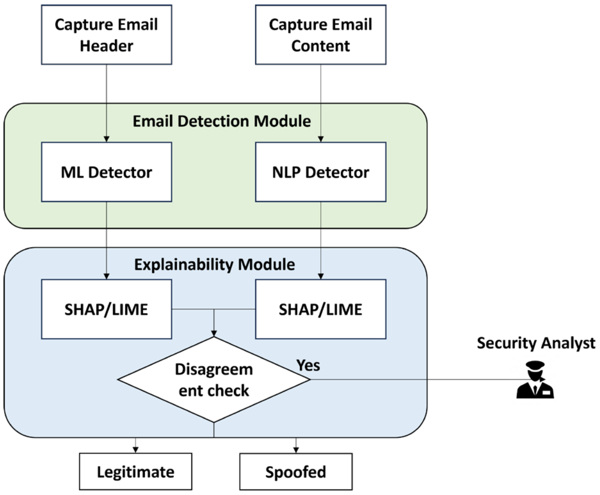

Fig. 5. Proposed Adversary-aware Framework
D. Experimental Settings

TABLE V
Experiment Server Specification

Specification	Properties
CPU	Intel(R) Core(TM) i7-8750H 2.20GHz x 12
Memory	GB 983.4
Operated System	Linux Ubuntu 18.04 LTS

E. Evaluation Metrics

The following evaluation metrics have been used to measure the models' performance: accuracy, recall, precision, and F1 score. These metrics, along with their descriptions, are defined in Table VI [40], [41], [42]. Also, the Attack Success Rate (ASR) of the framework components is calculated by measuring the model's accuracy when tested on a poisoned dataset [43]. A popular evaluation metric for binary classifiers, ROC-AUC, was considered as a performance measure of classification models rather than the F1 score and accuracy in this current paper, because of the following reasons:

- The selected datasets are unbalanced.
- It takes all decision thresholds into account when evaluating models.
- It has the slightest variance in evaluating individual models and the slightest variance in ranking a set of models.

Additionally, ROC-AUC has demonstrated several advantages over accuracy and other

TABLE VI
EVALUATION METRICS [23]

Metric	Description	Function
Accuracy	The ability of a classifier to correctly find spam/ non-spam	$\frac{TP + TN}{TP + FP + TN + FN}$
Recall	The ability of a classifier to correctly find spam	$\frac{TP}{TP + FN}$
Precision	The ability of a classifier to not misclassify spam	$\frac{TP}{TP + Fp}$
F1 Score	The harmonic mean of precision and recall	$\frac{2TP}{2TP + fp + FN}$

TABLE VII
THREAT MODEL

Adversary's 3D	Description
Goal	false negatives
Knowledge	perfect
Capabilities	Attack at training time (Causative)

TABLE VIII
ACCURACY OF THE FRAMEWORK ON ORIGINAL DATASETS

Datasets	Poison%	Poison Count	BERT	RF	BERT + RF
Spoofed Email	0.00%	0	0.5	0.53	0.51
Twitter Spam	0.00%	0	0.96	1	0.98

metrics. For example, it increases sensitivity in Analysis of Variance (ANOVA) tests, and decreases standard error when increasing sample size [44], [45].

F. Evaluation Results

The experiments conducted in this section focused on using the proposed framework to demonstrate the importance of designing an adversary-aware ML-based detector of spoofing email attacks. The adopted models were used to predict whether an email is spoofed based on 12 features. The same settings used by the author of the adapted models were followed. First, the datasets were processed to find missing values and explore the features. Then, the datasets were split into training and testing for model preparation. The ML-based detector was used for capturing the email's header, and the NLP-based detector was used for capturing the email's body. After building the models, a 10-fold cross-validation was used for evaluation. Table VIII shows the original accuracy (evaluation accuracy) of the framework on each dataset.

1) *Threat Models:* Threat modeling is an essential step towards identifying possible attack scenarios [47] ,[46] ,[22]. It helps define the goal, knowledge, and capability of an adversary. The adversary's goal can be based on the type of security violation, the target of the attack, and the specificity of the attack. For instance, the adversary's goal could be to compromise the integrity of an ML-based detector by manipulating a specific instance to cause an incorrect prediction. An adversary's level of knowledge about the targeted models varies and may range from perfect knowledge to limited knowledge or zero knowledge. An adversary's capability can enable him/her to either influence training data (causative attack) or testing data (exploratory attack).

2) *Adversarial Attack Scenarios:* Here, an experiment is discussed that illustrates a possible scenario of an adversarial attack against spoofed email detectors. Two causative and an evasion attack were launched against the proposed framework. One of the most common types of causative attack is a data poisoning attack, in which an adversary contaminates training datasets by either adding new samples or flipping existing ones, thereby degrading the performance of the learned model [48]. The adversary is assumed to have perfect knowledge of the targeted model; therefore, security by design is preferable over obscurity, and it can be considered the only viable mitigation method [49]. A label flipping attack, which is a type of data poisoning attack, was chosen for the experiment. In a label-flipping attack, an adversary changes the labels of some samples by flipping them to a different class. Additionally, this attack is categorized into untargeted and targeted attacks. In [50], it was shown that randomly flipping (i.e. untargeted attack) about 40% of the training data's labels decreased the prediction accuracy of the deployed classifier. However, many robust learning algorithms have been successfully developed to mitigate this attack [49], [51].

In this current study, both untargeted and

TABLE IX
Asr of Framework Detectors on the Spoofed Email Dataset With Different Poisoning%

Datasets	%Poison	Poison Count	BERT	RF	BERT + RF
Spoofed Email	Random 6.25%	50	0.5	0.497	0.51
Spoofed Email	Random 12.5%	100	0.5	0.497	0.5
Spoofed Email	Targeted 6.25%	50	0.5	0.493	0.5
Spoofed Email	Targeted 12.5%	100	0.5	0.491	0.5
Spoofed Email	Targeted 25%	200	0.5	0.483	0.5

TABLE X
Asr Of Framework Detectors on The Spoofed Email Dataset With Different Poisoning%

Datasets	%Poison	Poison Count	BERT	RF	BERT + RF
Twitter Spam	Random 3.59%	100	0.81	0.999	0.99
Twitter Spam	Random 12.5%	283	0.84	0.995	0.98
Twitter Spam	Targeted 3.59%	100	0.82	0.958	0.92
Twitter Spam	Targeted 12.5%	283	0.72	0.898	0.85

targeted attacks were considered, and different numbers of the dataset's labels were flipped, with the framework's performance being recorded. The threat model of the experiments is presented in Table VII. Additionally, to simulate the targeted label flipping, the SHAP explainable algorithm was used to select samples for flipping.

SHAP was used to find the most influential features, and based on the results, some instance labels were flipped. These two causative attacks (e.g., untargeted and targeted) simulate the best and worst-case scenarios, respectively. Table VIII shows the accuracy achieved by the framework detectors on the original datasets. Tables IX and X show the ASR obtained by the framework's detectors with different poisoning rates. Both results show that the attacker's success in manipulating the training data led to a degradation of the framework's accuracy with less than 7% poisoned data. The accuracy of RF dropped from 0.53 to 0.48 under the targeted attack for the spoofed email dataset. Whereas it dropped from 1 to 0.89 under the targeted attack for the Twitter spam dataset. The framework's overall prediction on the Spoofed Email dataset is low because of the dataset's quality. However, due to the lack of spoofed email datasets, we use it to simulate possible adversarial attacks against spoofed email detectors. The results show that as the percentage of contamination increases, accuracy decreases, especially in the case of targeted attacks. These findings are aligned with the conclusions of previous research [50].

3) Possible Defense Mechanism: Different defence mechanisms have been proposed to mitigate data poisoning attacks, such as sanitization [52], certifications [53], and randomization [54]. A common defence technique, sanitisation, where poisoned samples need to be identified and removed, was utilised in this current paper. The proposed framework contains XAI models to ensure adaptability to emerging attacks (i.e., monitor detectors' behavior) and ensure the explainability for debugging. The explainability of detectors on the original training datasets was recorded as shown in Figures 6. It shows the average SHAP importance for the spoofed class for RF on the original dataset. On the other hand, Fig 7 shows the average SHAP Feature Importance for RF using the Spoofed Email Dataset that includes 6.25% Poisoned Data via Targeted Label Flipping Attack. The IP reputation score, URLs, mismatched sender domain, SPF pass, DKIM pass, and DMARC pass were the features in order of importance

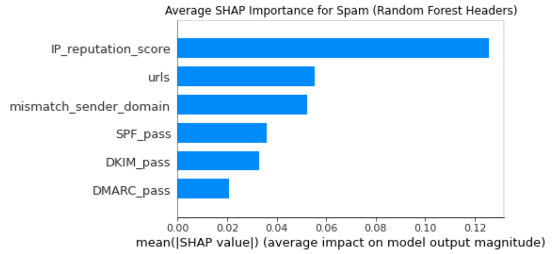

Fig. 6. Average SHAP Feature Importance for RF Using the Original Spoofed Email Dataset

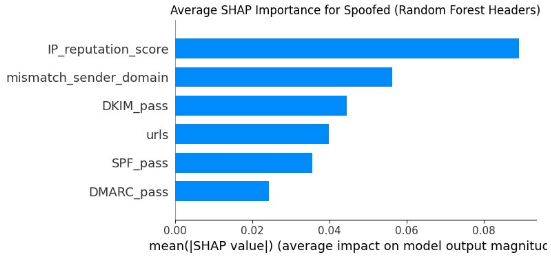

Fig. 7. Average SHAP Feature Importance for RF Using the 6.25% Targeted Attack

when using the original training dataset. Whereas, IP reputation score, mismatch sender domain, DKIM pass, URLs, SPF pass, and DMARC pass were the obtained feature importance order when adding 6.25% adversarial examples (i.e., Poisoned Data) into the training dataset. Also, Fig 8 and 9 show the average SHAP Feature Importance for RF using the original Twitter Spam Dataset, and the latter includes 12.5% Poisoned Data via Untargeted Label Flipping Attack, respectively. The results show that the feature importance of RF has changed, which can be used as an indicator of an apparent adversarial attack and requires further analysis by the framework admin.

IV. Discussion

The experiments presented in this paper demonstrate that considering the potential presence of an adversary who may attack the ML-based detector is crucial. Designing an ML-based detector for adversarial environments necessitates not only evaluating its performance against some adversarial attacks, but also considering how it can detect emerging attacks and how it can be debugged. The proposed framework was designed to detect potential adversarial attacks, such as label flipping, by monitoring both the accuracy and the disagreement between classifiers. Additionally, two XAI algorithms were employed for confirmation and debugging assistance.

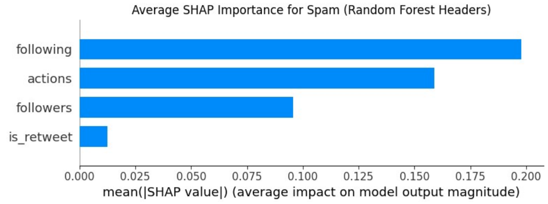

Fig. 8. Average SHAP Feature Importance for RF Using the Original Twitter Spam Dataset

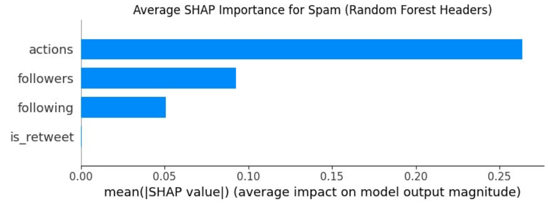

Fig. 9. Average SHAP Feature Importance for RF Using the 12.5% Untargeted Attack

A. Limitation

One limitation of this paper is that, due to the lack of a spoofed email dataset, only one dataset has been found. However, the quality of the dataset is poor, which affects the performance of the proposed framework. To overcome this limitation, a Twitter spam dataset was used that resembles a spoofed email dataset. Another limitation is that only the causative type of adversarial attack (i.e. poisoning attacks) was investigated against the proposed framework. Although XAI can play a crucial role in enhancing the trustworthiness, transparency, and security of ML-based models, its explainability can also be leveraged to compromise the system. It is essential to consider the vulnerability to cyberattacks for both the ML models and XAI algorithms deployed. Employing XAI increases the attack surface against ML-based detectors. Falsifying the explainability can be a target of an attacker. Adversaries can modify the explanation without affecting the model's prediction, which may cause a stockholder to make an incorrect decision [29].

Additionally, local and global metrics of the

LIME and SHAP algorithms were used to measure the XAI performance. Other metrics, such as fidelity and stability, can be used for confirmation. Fidelity or faithfulness is an essential metric in XAI. It measures how well an explanation reflects the model's actual behavior by focusing on the importance of different features. The Prediction Gap on Important feature perturbation (PGI) and the Prediction Gap on Unimportant feature perturbation (PGU) are examples of measures that can quantitatively assess the model's fidelity. On the other hand, stability, also known as robustness, is another crucial metric in XAI that measures the consistency of an explanation when the input data are slightly perturbed. Three submetrics can be used to calculate the stability: Relative Input Stability (RIS), Relative Output Stability (ROS), and Relative Representation Stability (RRS) [55], [56].

Another limitation of this study is that the robustness of the developed framework against only two potential adversarial attacks was evaluated. The literature has shown that several adversarial attacks can compromise ML-based detectors. Furthermore, the overhead of ML-based models, which refers to the computational resources, time, and complexity required to train, deploy, and run a machine learning model within the developed framework, can be regarded as an area for future work.

V. Conclusion

This paper provides a systematic survey of spoofed email detection techniques that utilise ML. Several related articles were critically analyzed to answer the research questions. The key research areas in spoofing email detection using ML algorithms were studied, including robustness against adversarial examples, adaptability to emerging adversarial attacks, and explainability. The SLR results indicated a lack of systematic literature reviews on spoofing email detection using ML techniques. Additionally, the results revealed that designing an adversary-aware ML-based detector for spoofed emails is rarely considered in the literature.

In response to the first research question, ML has been utilised for detecting spoofing email attacks based on analysing email headers, content or both. Also, the identified articles show that ML can be used to predict the likelihood that an individual will be a victim of spoofing attacks by evaluating their awareness. In addition, to answer the second research question, the SLR revealed that one article designed an ML-based detector that is robust against adversarial attacks, and one article considers both the robustness and adaptability. The explainability of ML-based detectors, which enables debugging of attacked ML models, has not been considered by the identified articles. To answer the third research question, the current study shows that designing ML-based spoofed email detectors for an adversarial environment is an open issue. Also, the robustness against the worst-case scenario, where the adversary has a high level of knowledge about the target, is another open issue.

Most importantly, the SLR shows that utilising XAI for designing an adversary-aware detector for spoofing email attacks has not been investigated in the literature. Consequently, an adversary-aware framework for detecting spoofed email was proposed. Multiple adversarial attack scenarios were performed to show the importance of designing an adversary-aware detector for spoofed email detection.

Additionally, an adversary-aware framework for detecting spoofed email was proposed. Two adversarial attacks were simulated to show the importance of considering the presence of adversities when designing ML-based detectors for adversarial environments. Additionally, a potential defence mechanism utilising XAI algorithms was proposed.

In the future, exploratory types of adversarial attacks, where an adversary targets an ML-based detector at the inference stage, will be used to evaluate spoofed email detectors. Additionally, research on text classification systems has demonstrated their vulnerability to adversarial examples. Examining adversarial attacks against spoofed email detectors is another area that requires investigation.

Funding

This article did not receive any specific grant from funding agencies in the public, commercial, or

not-for-profit sectors.

Conflict of Interest

The author declares that he has no conflict of interest.

References

[1] Fernandez, M. Korczynski, and A. Duda, “Early detection of spam´ domains with passive dns and spf,” in *International Conference on Passive and Active Network Measurement*. Springer, 2022, pp. 30–49.

[2] B. E. Nagarjuna, “*Artificial intelligence* for efficient spam and phishing email classification,” ARTIFICIAL INTELLIGENCE, vol. 13, no. 3, 2023.

[3] G. Giorgi, A. Saracino, and F. Martinelli, “End to end autorship email verification framework for a secure communication,” in *International Conference on Information Systems Security and Privacy*. Springer, 2020, pp. 73–96.

[4] S. Shukla, M. Misra, and G. Varshney, “Forensic analysis and detection of spoofing based email attack using memory forensics and machine learning,” in *International Conference on Security and Privacy in Communication Systems*. Springer, 2022, pp. 491–509.

[5] P. N. Wosah, “A framework for securing email entrances and mitigating phishing impersonation attacks,” *arXiv preprint arXiv:2312.04100*, 2023.

[6] G. Kamal and M. Manna, “Detection of phishing websites using na¨ıve bayes algorithms,” *International Journal of Recent Research and Review*, vol. 11, no. 4, pp. 34–38, 2018.

[7] A. Al-Sinayyid, M. J. A. Jewel, V. Mannuru, and K. Sasidhar, “Defending characteristics and attribution analysis for phishing attacks,” in *2023 International Conference on Computational Science and Computational Intelligence* (CSCI). IEEE, 2023, pp. 868–874.

[8] Y. T. Goh, “Phishing email detection using machine learning,” 2021.

[9] C. S. Jalda, A. K. Nanda, and R. Pitchai, “Spoofing e-mail detection using stacking algorithm,” in *2022 8th International Conference on Smart Structures and Systems (ICSSS)*. IEEE, 2022, pp. 01–04.

[10] K. Evans, A. Abuadbba, T. Wu, K. Moore, M. Ahmed, G. Pogrebna, S. Nepal, and M. Johnstone, “Raider: Reinforcement-aided spear phishing detector,” in *International Conference on Network and System Security*. Springer, 2022, pp. 23–50.

[11] M. F. Ansari, A. Panigrahi, G. Jakka, A. Pati, and K. Bhattacharya, “Prevention of phishing attacks using ai algorithm,” in *2022 2nd Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON)*. IEEE, 2022, pp. 1–5.

[12] H. Gascon, S. Ullrich, B. Stritter, and K. Rieck, “Reading between the lines: content-agnostic detection of spear-phishing emails,” in *Research in Attacks, Intrusions, and Defenses: 21st International Symposium, RAID 2018, Heraklion, Crete, Greece, September 10-12, 2018, Proceedings 21*. Springer, 2018, pp. 69–91.

[13] T. Krause, R. Uetz, and T. Kretschmann, “Recognizing email spam from meta data only,” in 2019 I*EEE Conference on Communications and Network Security (CNS)*. IEEE, 2019, pp. 178–186.

[14] O. Odunibosi, “Classification of email headers using random forest algorithm to detect email spoofing,” Ph.D. dissertation, Dublin, National College of Ireland, 2019.

[15] P. Boyle and L. A. Shepherd, “Mailtrout: a machine learning browser extension for detecting phishing emails,” in *34th British HCI Conference*. BCS Learning & Development, 2021, pp. 104–115.

[16] S. Heena et al., “Online correspondence hard sell ranking using expert system and development of’thinking’computer systems,” *Mathematical Statistician and Engineering Applications*, vol. 68, no. 1, pp. 129–140, 2019.

[17] Y. Fang, Y. Yang, and C. Huang, “Emaildetective: An email authorship identification and verification model,” *The Computer Journal,* vol. 63, no. 11, pp. 1775–1787, 2020.

[18] P. Wu and H. Guo, “Holmes: An efficient and lightweight semantic based anomalous email detector,” in *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2022, pp. 1360–1367.

[19] A. Dalton, E. Aghaei, E. Al-Shaer, A. Bhatia, E. Castillo, Z. Cheng, S. Dhaduvai, Q. Duan, M. M. Islam, Y. Karimi et al., “The panacea threat intelligence and active defense platform,” *arXiv preprint arXiv:2004.09662*, 2020.

[20] B. Biggio and F. Roli, “Wild patterns: Ten years after the rise of adversarial machine learning,” vol. 84. Elsevier, 2018, pp. 317–331.

[21] N. Dalvi, P. Domingos, Mausam, S. Sanghai, and D. Verma, “Adversarial classification,” in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’04. New York, NY, USA: ACM, 2004, pp. 99–108.

[22] B. Biggio, G. Fumera, and F. Roli, "Security Evaluation of PatternClassifiers under Attack," *Knowledge and Data Engineering*, , vol. 26, no. 4, pp. 984–996, 2014.

[23] N. Imam, "Adversary-aware, machine learning-based detection of spam in twitter hashtags," Ph.D. dissertation, University of York, 2021.

[24] T. S. Sethi and M. Kantardzic, "Handling adversarial concept drift in streaming data," *Expert Syst. Appl.*, vol. 97, pp. 18–40, 2018.

[25] B. Biswas, A. Mukhopadhyay, A. Kumar, and D. Delen, "A hybrid framework using explainable ai (xai) in cyber-risk management for defence and recovery against phishing attacks," *Decision Support Systems*, vol. 177, p. 114102, 2024.

[26] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[27] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining,* 2016, pp. 1135–1144.

[28] A. Fan, Y. Jernite, E. Perez, D. Grangier, J. Weston, and M. Auli, "ELI5: Long form question answering," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum, and L. Marquez, Eds.` Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3558–3567. [Online]. Available: https://aclanthology.org/P19-1346

[29] N. H. Imam, "Adversarial examples on xai-enabled dt for smart healthcare systems," Sensors, vol. 24, no. 21, p. 6891, 2024.

[30] S. Arya and S. Chamotra, "Multi layer detection framework for spearphishing attacks," in *Information Systems Security: 17th International Conference, ICISS 2021, Patna, India, December 16–20, 2021, Proceedings 17*. Springer, 2021, pp. 38–56.

[31] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering– a systematic literature review," *Information and software technology*, vol. 51, no. 1, pp. 7–15, 2009.

[32] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 2, pp. 590–611, 2023.

[33] B. Naqvi, K. Perova, A. Farooq, I. Makhdoom, S. Oyedeji, and J. Porras, "Mitigation strategies against the phishing attacks: A systematic literature review," *Computers & Security*, p. 103387, 2023.

[34] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE access*, vol. 7, pp. 167653–167671, 2019.

[35] F. Huseynov and B. Ozdenizci Kose, "Using machine learning algorithms to predict individuals' tendency to be victim of social engineering attacks," *Information Development*, vol. 40, no. 2, pp. 298–318, 2024.

[36] N. Chanthati, "How the power of machine–machine learning, data science and nlp can be used to prevent spoofing and reduce financial risks," *Global Journal of Engineering and Technology Advances*, vol. 20, no. 2, pp. 100–119, 2024.

[37] A. I. Champa, M. F. Rabbi, and M. F. Zibran, "Curated datasets and feature analysis for phishing email detection with machine learning," in *2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI)*. IEEE, 2024, pp. 1–7.

[38] N. Alshahrani, S. Alshahrani, E. Wali, and J. Matthews, "Arabic synonym BERT-based adversarial examples for text classification," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop,* N. Falk, S. Papi, and M. Zhang, Eds. St. Julian's, Malta: Association for Computational Linguistics, Mar. 2024, pp. 137–147. [Online].Available: https://aclanthology.org/2024.eacl-srw.10/

[39] T. S. Sethi and M. Kantardzic, "Handling adversarial concept drift in streaming data," *Expert systems with applications*, vol. 97, pp. 18–40, 2018.

[40] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *arXiv preprint arXiv:2010*.16061, 2020.

[41] G. Lin, N. Sun, S. Nepal, J. Zhang, Y. Xiang, and H. Hassan, "Statistical twitter spam detection demystified: performance, stability and scalability," *IEEE access,* vol. 5, pp. 11142–11154, 2017.

[42] N. H. Imam and V. G. Vassilakis, "A survey of attacks against twitter spam detectors in an adversarial environment," Robotics, vol. 8, no. 3, p. 50, 2019.

[43] E. Nowroozi, N. Jadalla, S. Ghelichkhani, and A. Jolfaei, "Mitigating label flipping attacks in malicious url detectors using ensemble trees," *IEEE Transactions on Network and Service Management,* 2024.

[44] K. Sauka, G.-Y. Shin, D.-W. Kim, and M.-M. Han, "Adversarial robust and explainable network intrusion detection systems based on deep learning," *Applied*

Sciences, vol. 12, no. 13, p. 6451, 2022.

[45] J. Li, “Area under the roc curve has the most consistent evaluation for binary classification,” *PloS one*, vol. 19, no. 10, p. e0307998, 2024.

[46] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, “The security of machine learning,” *Mach. Learn.,* vol. 81, no. 2, pp. 121–148, 2010.

[47] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar, “Can machine learning be secure?” in *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security*, ser. ASIACCS ’06. New York, NY, USA: ACM, 2006, pp. 16–25.

[48] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, “The security of *machine learning,*” Machine learning, vol. 81, pp. 121–148, 2010.

[49] E. Rosenfeld, E. Winston, P. Ravikumar, and Z. Kolter, “Certified robustness to label-flipping attacks via randomized smoothing,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 8230–8241.

[50] B. Biggio, B. Nelson, and P. Laskov, “Support vector machines under adversarial label noise,” in *Asian conference on machine learning*, 2011, pp. 97–112.

[51] L. Lavaur, Y. Busnel, and F. Autrel, “Systematic analysis of labelflipping attacks against federated learning in collaborative intrusion detection systems,” in *Proceedings of the 19th International Conference on Availability, Reliability and Security*, 2024, pp. 1–12.

[52] G. F. Cretu, A. Stavrou, M. E. Locasto, S. J. Stolfo, and A. D. Keromytis, “Casting out demons: Sanitizing training data for anomaly sensors,” in *2008 IEEE Symposium on Security and Privacy (sp 2008).* IEEE, 2008, pp. 81–95.

[53] J. Steinhardt, P. W. W. Koh, and P. S. Liang, “Certified defenses for data poisoning attacks,” *Advances in neural information processing systems*, vol. 30, 2017.

[54] J. Lee, Y. Cho, R. Lee, S. Yuk, J. Youn, H. Park, and D. Shin, “A novel data sanitization method based on dynamic dataset partition and inspection against data poisoning attacks,” *Electronics*, vol. 14, no. 2, p. 374, 2025.

[55] K. N. Pellano, I. Strumke, and E. A. F. Ihlen, “From movements to¨ metrics: Evaluating explainable ai methods in skeleton-based human activity recognition,” Sensors, vol. 24, no. 6, p. 1940, 2024.

[56] D. Alvarez-Melis and T. S. Jaakkola, “Towards robust interpretability with self-explaining neural networks,” in *Advances in Neural Information Processing Systems*, 2018, pp. 7786–7795.