TESI DI DOTTORATO

UNIVERSITÀ DEGLI STUDI DI NAPOLI "FEDERICO II"

DIPARTIMENTO DI INGEGNERIA ELETTRICA
E TECNOLOGIE DELL'INFORMAZIONE

DOTTORATO DI RICERCA IN
INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

A MACHINE AND HUMAN LEARNING APPROACH FOR PHISHING DEFENSE

Design, Implementation and Evaluation of
a collaborative framework against Email Attacks

LUIGI GALLO

Il Coordinatore del Corso di Dottorato	Il Tutore
Ch.mo Prof. Daniele RICCIO	Ch.mo Prof. Alessio BOTTA

A. A. 2021–2022

“Tieni sempre
conto del cambio
di variabili”

Acknowledgments

I want to express my sincere gratitude to the two entities that made this work possible and supported me every step of the way. I would like to thank the University of Napoli “Federico II”, in the person of Prof. Alessio Botta, who represents a professional and personal guide for me, an example to follow for life. I am grateful to Telecom Italia and the Cyber Security Lab professionals for accompanying me on this journey by devoting time and resources.

Contents

List of Figures

Introduction and Motivation

Email is still one of the most used channels for making cyber attacks, thus exposing companies to frequent attempts at security breaches. This collides with the importance and the widespread use emails have in everyday work life. The Internet Security Threat Report by Symantec [1] states that spam level is on the rise, as it has been every year since 2015, with 55 percent of emails received in 2018 being categorized as spam. Within the category of spam emails fall either innocuous attempts to market and sell products, or messages that contain significant threats, such as phishing attempts to steal credentials or malware delivery for espionage and theft of sensitive data. The upward trend is also confirmed for phishing attacks, with a 70% increase in the 2021 yearly period [2]. In 2016, the FBI raised the alarm over this important problem as this kind of attacks increased in number and malignance [3], and in 2019 a monetary loss in USA of about U.S. $1.8 billion has been estimated with just a subset of attacks feasible with emails [4]. The European Cybercrime Centre (EUROPOL) claims that email attacks are used as the primary infection vector in 78% of cyber espionage incidents [5]. With the coronavirus pandemic shifting daily activities online in many parts of the world, the cybercrime situation has even worsened. As a consequence, company SOCs (Security Operation Center) and CERTs (computer Emergency Response Team) need large teams of security analysts, typically named Anti-Phishing groups[1], monitoring this specific type of threats. Unfortunately, the problem of email security threats is more and more challenging because of the really huge number of spam emails across the network every day, among which malicious emails are mixed. This makes the work of analysts a real "needle in a haystack" search.

Techniques for building effective spam emails vary from using advanced strategies to escape spam filters to sophisticated social engineering techniques

[1] Due to a common abuse of notation, "Phishing" means both a certain subclass of attacks and in some cases also the whole set of attacks via email. Hence the name "Anti-phishing group".

to trick people. While spam filters work well as a countermeasure to some troubles caused by spam such as network overload, loss of time and productivity, irritation and discomfort, they still lack to solve the problem of email as an attack vector. The spectrum of email attacks is varied, ranging from the legacy ones concerning purely technical aspects (e.g. sender address spoofing), still feasible due to SMTP configurations and vulnerabilities [6, 7], to the more sophisticated socio-technical methods made possible by modern machine learning and social engineering techniques. The aim of the attackers in these scenarios is usually to spread malware, steal authentication credentials, or commit financial fraud. Depending on the goal, the attacks can be classified as: malware propagation, (spear) phishing, (CEO) fraud, and scam. The most dangerous ones are 'tailored' against specific organizations or groups of people, and differ significantly from generalist attacks. Employees of big companies are normally trained not to be fooled by email attack attempts; however, despite this training, employees usually fall for spam mails for various reasons, including: the fact that large companies have employees of all age ranges, with various education degrees and different technology expertise and the lack of concentration of people to recognize phishing attacks can also be crucial [8]. Every single employee can represent a point of entry for spammers and attackers. In the context of a company with tens of thousands of employees, millions of emails are received every day, 55% of which are unsolicited [1]. According to the estimates of the security group, while 95% of these unsolicited mails are blocked by spam filters, the remaining 5% (about 25 thousand every day) is still a potentially dangerous amount of emails, too large to monitor and control.

In this work a spam email is simply an unwanted email, and the security analyses are not concerned with most of them. It is important to understand if any of them created or has the potential to create a *security incident*: "a security-relevant system event in which the system's security policy is disobeyed or otherwise breached" [9]. Only this (small) portion of spam e-mails is of interest, from a security point of view. When an employee browses a malicious website or downloads a malicious email attachment (e.g. ransomware, trojan etc.), a security incident can occur. Security incidents can have different impacts, depending on the number and role of the employees involved, the nature of the threat, and how effective the security systems (i.e. corporate antivirus) are against them. We call *critical spam* the emails that caused or have the potential to cause a security incident. Since the number of unsolicited emails received by large companies is indeed huge and constantly increasing, their manual analysis is not feasible. An automatic mechanism to detect criti-

cal spam that bypasses common antispam filters becomes therefore necessary.

This thesis reports on the activities performed during the last 3 years in the anti-phishing group of TIM (the biggest Italian telco, also known as *Telecom Italia*, which has supported the work) that comprise: construction of a system for real spam emails collection; labeling of this data; study of the characteristics of critical spam emails; design, development, and deployment of an automated system for critical spam detection based on machine learning techniques; conduction of a data-driven awareness campaign based on insights derived from the previous activities. On average, 30 million emails per month reach the 100,000 mailboxes of the company employees and external collaborators, most of which are filtered by the spam filter. The starting idea was to collect over the years spam emails that pass the spam filter and are reported by users as unwanted, also storing the information produced by the SOC analyst about the possible security incident occurred. To this aim, a collaborative framework for reporting and monitoring of such spam emails has been designed also with the goal of collecting data (Chapter 3). This framework supports the work of security analysts, allowing them to annotate the results of their analysis directly on the data, thus obtaining a solid ground truth (Chapter 3.1). With this approach, a labeled dataset of 22,000 unique emails reported in the last 2 years has been collected. Several legacy and novel features have been extracted from the samples of the dataset (Chapter 3.2). Various machine learning algorithms have been used to perform a binary classification: critical or not relevant spam. The main classification algorithms based on machine learning have been tested and compared in order to find the best one, including Gaussian Naive Bayes, Decision Trees, Support Vector Machines, Neural Networks and Random Forest (Chapter 4). The feature ranking work provides information on how critical emails are built and can be detected. This knowledge led to the design of a week-long awareness campaign, which involved all 40,000+ employees of the company, including top managers and executives (Section 4.5).

Working on this field for a few years, and according to the current literature [10, 11], it has become clear that there is an urge to understand in detail the cognitive aspect of the phishing phenomenon, to design practical and effective solutions. As computer systems become more and more secure from a technical viewpoint, attackers increasingly target the human behind it. Improving the security awareness of Internet users is now a matter of paramount importance. At least one person clicked a phishing link in around 86% of organizations, potentially causing a security incident, according to Cisco 2021 CyberSecu-

rity Threat Trends Report [12]. In phishing attacks, victims are lured through false correspondence that lead to carefully constructed phishing websites. The engineering of phishing e-mails is, in fact, the main concern for phishers and studying the different strategies that can be employed is fundamental to have a thorough understanding of the problem. In typical scenarios, phishers apply several techniques to increase the *credibility* of their messages, such as imitating legitimate communications from knowledgeable companies and resort to strategies to deceive the victim's perception. For example, they may generate an e-mail characterized by a layout that is similar to the one of an e-mail sent by the real company, imitating its text, images, and overall appearance; or they may use different techniques to pass off a malicious link as a legitimate one, which are usually referred to as domain-squatting techniques, to persuade the victims to click on a link in the body of the e-mail. On the other hand, attackers usually implement strategies that influence human decision making, increasing the probability that the victim complies with the request of the e-mail. These strategies involve an artful shaping of the textual content of the e-mail, such as the inclusion of sentences that express a *Persuasion Principles*, with the goal of exploiting one or more *Cognitive Vulnerabilities* of the victim [13]. On a general note, the engineering of phishing e-mails has evolved a lot in the past years and is expected to keep doing so in the upcoming ones. It is therefore fundamental to understand why each person falls for phishing to devise more effective training processes.

In the final part of this work, a system was developed to test user knowledge about phishing and to collect and share data on this important issue. Through the system users interact with a series of e-mails which can be either a representation of a phishing attack or a legitimate communication and have to deem each one as 'phishing' or 'legitimate'. Before the test, users are asked to insert anonymous information about their background. Data collected can be used for several purposes, including the analysis of the awareness level of users with relation to their background and to the characteristics of the e-mails. The goal of this research is also to collect and analyze data on the ability of users to recognize phishing e-mails, the effectiveness of different attack strategies, both cognitive and technical, the behavior of users with similar characteristics (e.g., educational or psychological), and the effectiveness of specific attack strategies on users with specific characteristics. This information can be used to design solutions to phishing that take into consideration the individual characteristics of users and protect them against the typologies of attacks that are more likely to deceive them. These solutions include ergonomic e-mail clients

that provide tailored notifications to vulnerable subjects when a suspect e-mail is received and awareness campaigns specialized on the specific cognitive vulnerabilities of each user. The system and the data are publicly accessible at ***https://spamley.comics.unina.it/***.
To recap, the main contributions of this thesis are:

- A broad representation of the technical background behind the world of phishing attacks and a comprehensive review of the scientific literature, presented in Chapter 1 and 2. From this, the current difficulty in dealing with phishing attacks is evident, which motivates the work of the following contributions.

- The construction of a system for real spam emails collection, manually labelled as 'Critical' or 'Not_relevant', is presented in Chapter 3. Considering this raw data, a feature set comprising both traditional and novel features has been designed (Section 3.2).

- The design, development, and evaluation of an automated system for critical spam detection based on machine learning techniques is presented in Chapter 4. This includes an important explanation on how critical emails are built and can be detected (Section 4.3), and the conduction of a data-driven ethical phishing campaign based on insights derived from the previous activities (Section 4.5).

- The design and deployment of a system with the dual aim of disseminating awareness among users and collecting and sharing data regarding user behavior when reading e-mails, presented in Chapter 5. The focus of scientific work on phishing has shifted from more technical to more human-related aspects. The collective knowledge of different studies on phishing has been actualized to make the e-mails featured in the system as realistic and accurate as possible. This contribution includes also the analysis that is possible to perform on the data collected through the system (Section 5.5).

This thesis is based on the following peer-reviewed publications:

- Luigi Gallo, Alessandro Maiello, Alessio Botta, Giorgio Ventre, *2 Years in the anti-phishing group of a large company*, Computers & Security, Volume 105, 2021, 102259, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2021.102259. [14]

- Luigi Gallo, Alessio Botta, and Giorgio Ventre. 2019. *Identifying threats in a large company's inbox*. In Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks (Big-DAMA '19). Association for Computing Machinery, New York, NY, USA, 1-7. https://doi.org/10.1145/3359992.3366637 [15]

Chapter 1

Background on Spam and Phishing emails

Also known as spamming, **Spam** is the activity of sending unsolicited communications without the recipients providing their address or consenting to receive messages in their inboxes. Spam is usually sent by email, but recently it has also taken the form of SMS text messages, comments/chat messages on social networks or instant messaging apps. The term *"spam"* has not always been linked to the digital world. Spam originated as an acronym of the words 'Spiced' and 'Ham', and SPAM was the brand name of a well-known spiced canned meat that was very popular in England after the Second World War. Only in 1970 the term *"spam"* took on a meaning similar to its current one. It was in this year that a particular episode of the much-loved BBC series *Monty Python's Flying Circus* featured a comedy sketch set in an inn whose menu consisted entirely of spam. The waitress 'recited' the single-issue menu, explaining the virtues of spam, while in the background, a strange chorus of characters dressed as Vikings repeated the fateful word, 'Spam, spam, spam!', completely drowning out the conversations of the other patrons. Since then, the term 'spam' has come to be associated with the concept of unwanted, annoying, excessive and obsessive information, capable even of overriding, and to some extent obscuring, any kind of useful communication [16]. The use of the term "spam" in its new meaning (unsolicited or unwanted e-mails, "junk" e-mails) was first introduced in 1993. The Usenet administrator Richard Depew devised a program which mistakenly caused dozens and dozens of identical messages to be sent within the newsgroup on 31 March 1993. The recipients of these persistent and inappropriate messages referred to them as "spam".

The first mass spam mailing dates back to 1994, when the law firm Canter & Siegel used the Usenet network to disseminate advertising messages in large quantities offering services of a dubious nature. This event marked the very beginning of commercial spam. Today, after almost three decades, spam has taken on impressive proportions: it accounts for more than 50% of the total volume of global e-mail traffic, causing the problems outlined in the Section 1.2 . The reason why unsolicited communications have found a powerful 'ally' in e-mails is that it is enough to know, or even try to guess, an e-mail address in order to send a message without any particular cost, effort or authorization. In addition, people normally distribute their e-mail address in order to be contacted, just as they do with the home address in order to receive non-electronic mails, with the difference that the latter cost money and are not scalable. Section 1.1 explains the techniques used by spammers to send mass e-mails.

1.1 Techniques used by spammers

Spam has become an impressive phenomenon: it accounts for more than 50% of the total volume of global e-mail traffic [1]. In order to make such mass mailings possible, and to make sure that they are not detected by spam filters and other security systems, which we will discuss later, spammers use specific technologies and programs, and invest time and money in their implementation. The main activities needed to set up an effective spam campaign are the following:

- Harvesting and verification of email addresses, and eventual classification by category of interest [17, 18].
- Preparation of the platforms necessary to conduct the mass mailing [19, 20].
- Creation of the software necessary to carry out the mass mailing [21].
- Creation of the advertising messages to be sent [22].

The different activities described above can also be carried out by different actors, who activate actual 'partnerships' between them on the basis of mutual interests [23]. It is a real illegal industry [24] generating millions of dollars in revenue [25, 26] .

Harvesting and verification of email addresses. Spammers use databases of e-mail addresses that contain not only the actual address, but also additional information such as the geographic area and sphere of interest of the target. The email addresses in these databases are collected in different ways: randomly generated as a combination of proper names, digits, and 'buzzwords'; obtained by scanning publicly accessible information sources, such as websites, forums, chats, message boards, and so on, looking for combinations of particular words such as 'word1@word2.domain'; extracted from online service databases, leaked on the internet or stolen; stolen from users' personal data, by means of computer viruses and other malware [17, 18]. Email addresses obtained through these means are verified to assess their existence and activity. The most commonly used methods are as follows:

- Sending a test message: by sending a random message that can pass the spam filters, and analysing the answer given by the mail server (accepted or rejected message), it's possible to understand if the address is active or not.
- Inserting an "unsubscribe" link: if the user clicks on the link in an attempt to unsubscribe from the service, it's possible to check the validity of the address and even if the user is actively using it.
- Inserting a link in the message to an image on a specific web server: when opening the spam message, the mail client, if designed to do so, will automatically download the image. In this case, the owner of the website will be sure that the e-mail address is valid and that the spam message has been read.

None of the methods are foolproof, so there will always be a percentage of invalid addresses in the databases.

Preparation of the platforms necessary to conduct the mass mailing. The three main methods to perform mass mailing are as follows:

- Direct mailing through rented servers: due to current anti-spam systems based on blocklisting, this technique can only be used towards users using email services that do not use blocklists. In any case, it is necessary to have a very large set of servers that can be continuously renewed.
- Use of open relay and open proxy: an 'open relay' is a mail server that allows any user to send an email message to any email address. Al-

though almost all of these servers are properly filtered by security systems, some are still in operation and are still used.

- Use of users' computers, through malware installed in advance.

Nowadays, most spam mailings are conducted using botnets, i.e. networks composed of large numbers of infected computers controlled by spammers [19, 20]. To hack into users' computers and take control of them, the rogues use various methods: Trojan programs embedded in pirated software and distributed via file-sharing networks; exploiting bugs and vulnerabilities in operating systems and popular software; using worms, viruses, or other malware distributed via e-mail.

Creating the software necessary to perform the mass mailing. In order for mass mailing to be effective, it is necessary that all the emails in the same campaign are sent quickly, before the spam filters are reconfigured or the databases they refer to are updated. The programs used by spammers are not only able to send a large amount of e-mails in a very short time, but also have many other functionalities to bypass security systems:

- allow spam emails to be sent either through 'open' services (mail relays, proxy servers) or through botnets;
- allow to diversify spam messages by inserting additional dynamic texts;
- allow message headers to be masked, making it more difficult to detect them as spam;
- allows monitoring the validity of email address databases;
- allow to verify the correct reception of the message, and, if not, to resend it through different methods to bypass possible blocklists.

These mechanisms are adequately engineered, with even very sophisticated orchestration techniques [21].

Creation of advertising messages to be sent. Current anti-spam systems, as described in next sections, are able to recognise and filter out identical or particularly similar spam messages. In order to bypass these filters and deliver spam to victims' inboxes, spammers use various techniques to make messages belonging to the same campaign as 'unique' as possible. The most common are:

- Introducing random text, hidden text, and smear: Spammers can place a musical text, an excerpt from a classical text, a poem or any random set of words at the beginning or end of the message. In addition, using the tools provided by HTML, it is possible to insert 'invisible' text, e.g. written in extremely small fonts, or using the same color as the background, or simply inserting text into the code with the feature of not showing it on the screen.

- Hiding text in an image: spam messages can also be delivered in the form of images that may contain not only pictures of products and brands, but also text, making it difficult for spam filters to automatically analyse the message.

- Image fragmentation: the image containing the text and appearing to the user as "whole" can be broken up into several fragments.

- Text paraphrase: Spam messages belonging to the same campaign differ from each other while conveying the same advertising message. Each of them presents a different but logical and coherent version of the same text, making it difficult to recognise that they belong to the same campaign automatically.

All these techniques are implemented by the software that spammers use to create and disseminate spam messages [22].

1.2 The harms of spam

Initially characterised by advertising mailings of rather modest proportions, the spam phenomenon has evolved and expanded, becoming a serious technical, economic and even social threat. These are just some of the negative effects of spam:

- Network overload: the huge volume of traffic caused by spam forces ISPs to over-provision their networks, and companies to incur huge costs to maintain the e-mail servers that are forced to receive and process these streams of unwanted messages [27]. Malicious users can also use the SMTP server to perform *Denial-of-Service (DoS) attacks.* This basically means flooding other servers with a huge amount of emails to affect their performance or even cause a crash. DoS can also be used to flood an inbox to hide any warning messages about possible security breaches.

- Loss of time and productivity: when spam gets past spam filters and reaches the end-user's e-mail box, the end-user will be forced to manually clean the personal e-mail box of such messages, taking time away from normal productive activities. Moreover, it will be difficult for the user to find important emails if they are mixed with dozens of spam emails.

- Irritation and discontent: not only is spam made up of unsolicited and unwanted emails, but sometimes the content can be offensive or in bad taste, causing unnecessary irritation to the user.

- Vehicle for attacks: unsolicited e-mails are often used to launch cyber-attacks to spread malware, steal credentials or perform financial fraud. This is the main problem with spam emails that motivates this thesis, as spam filters fail to provide a solution. In a fairly generalised manner, unsolicited emails that have a malicious intent are called phishing emails. It is difficult to assess the damage caused by phishing attacks to banks, companies and institutions because they do not tend to disclose such information. In addition, many users are unwilling to acknowledge that they have fallen victim to phishing attacks. This happens out of fear of humiliation, financial loss or legal liability. Studies [3, 4, 28, 5] estimate the damage of direct losses to victims in the United States and in Europe reaching even several billions $/€ per year. In addition phishing attacks are also a major problem in terms of trust among users and the misuse of email as a means of communication.

1.3 Phishing attacks

In the past, cyber attacks were described as a series of computer operations, entirely related to technology and directed to it, the execution of which did not require any kind of interpretation, it was exclusively the result of logical steps contained in instructions written in a particular programming language. Consequently, all security research efforts were focused on securing the systems and their communication protocols, producing the culture of designing new systems/protocols with the *security-by-design*. However, in the cyber security chain there is probably the weakest link left uncovered: the human factor. As systems have become increasingly secure (e.g. fewer and less exploitable vulnerabilities), attacks have evolved into real social attacks, exploiting people's

cognitive vulnerabilities rather than system vulnerabilities. The set of fraudulent cognitive-based techniques finalized to steal data and account credentials, propagate malware and commit financial frauds is defined as **Phishing**[1].

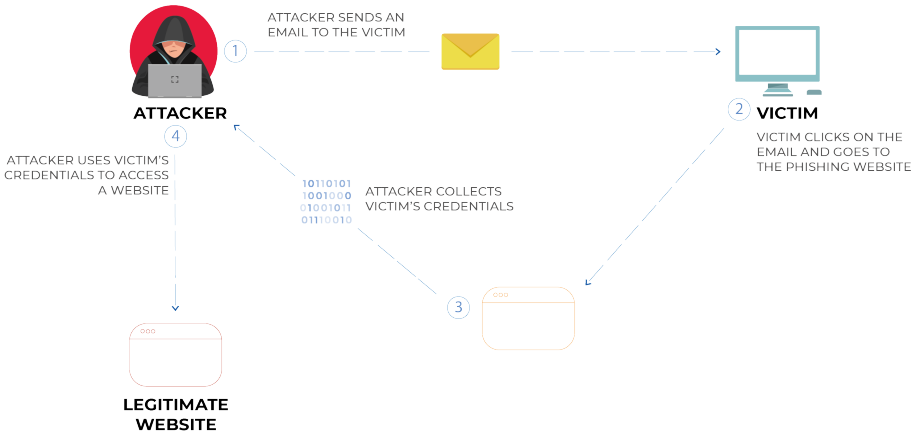


Figure 1.1: Phishing attack workflow

Exploiting *social engineering* techniques aimed at increasing the credibility and persuasiveness of the message (step 1 in Figure 1.1), the phisher tends to acquire unsuspecting users' identity or sensitive confidential information through the use of spoofed emails, fake websites, dubious online advertisements/promotions, fake SMS from service providers or online companies, spear phishing etc. Common targets in social engineering schemes include large corporations, financial institutions, payment companies, military and government agencies that have usually suffered tremendous financial and brand credibility damage [29, 30].

The main means by which phishing attacks are carried out is email, hence **phishing emails**. This is motivated by the following facts:

- Email is a fundamental method of communication in the world and everyone (from companies to official bodies), rely on email in their daily

[1] According to some theories, the etymology of the word phishing comes from an adaptation of the English verb *to fish*, evoking the image of "fishing for users on the Internet"; for others, instead, it is the result of the fusion of the English words *(p)assword*, *(h)arvesting* and *(f)ishing*, indicating the collection of keywords and access codes to economic-financial services; finally, for others, it would derive more simply from the union of the terms *(ph)reaking* (activity of study, experimentation or exploitation of telephones, telephone companies and systems that make up or are connected to the general telephone network) and *f(ishing)*

operations. It's easy and comfortable. We all use email, including non-computer experts, so we are all potential victims of phishing emails. In addition, we have already described in Section 1.1 how it is easy to be reached by an unsolicited e-mail.

- The e-mail service is one of the oldest on the Internet, and the Simple Mail Transfer Protocol (SMTP) designed in 1982 is still universally used: all e-mail sent over the Internet is transferred using this outgoing protocol. Obviously it was not designed with the *security-by-design* approach, therefore has a major security limitation: the recipient of an e-mail cannot 'authenticate' the sender with certainty, so the recipient cannot know with certainty whether the sender of a message is really who the sender claims to be. This allows the execution of *impersonation attacks*, where the attacker makes the email message appear to come from a trusted source with a seemingly legitimate address, for example pretending to be the CEO of a company (e.g *steve.jobs@appl3.com*) or a close friend of the recipient. This aspect is discussed in more detail below.

- In some configurations, the SMTP is affected by certain vulnerabilities, for example if no DMARC policy is set *spoofing attack* is possible: the attacker fakes the email header so that the email client displays the fraudulent sender's address, which most users take for granted, (e.g., the sender's email address is *fraudster@cybercrime.com*, but the recipients see *steve.jobs@apple.com* in their inbox). These attacks are still widespread [6, 7].

In conjunction with social engineering, to make phishing emails credible, criminals often use several techniques such as link manipulation, filter evasion and psychological vectors.

Link Manipulation. The emails proposed to the victims, are generated so that the domains used are plausible. Previous publications [31] have studied several attack techniques for this purpose, which are often referred to as *domain squatting* techniques: *registration of domains corresponding to well-known brands or famous people, with the clear purpose of deriving economic advantage*. Variants include *typosquatting*, in which an attacker creates a domain that differs from a known domain only by a typical typo, for example *paypl.com* or *paypaal.com* [32]. Alternatively, a new technique that

replaces typosquatting is *combosquatting*. Combosquatting involves modifying a known domain by adding terms in such a way that the resulting domain is still credible, e.g., *bankofamerica-security.com* or *secure-paypal.com* [33]. Furthermore, a domain cannot only contain Latin letters, but also letters from other alphabets. Therefore, an attacker may replace one or more characters in a known domain with similar characters from other alphabets, for example, *bankofámerica.com*, which is referred to as a *homograph domain* [34]. Registering a domain that sounds similar to a known domain, however, is called *soundsquatting*, e.g., *guaranty-bank.com* instead of *guarantee-bank.com* [35].

Filter Evasion. Phishers use visual deception tricks to mimic legitimate text, images and windows. Sometimes, images are used in place of text to make it more difficult for anti-phishing or anti-spam filters to detect the text commonly used in these emails. It is possible, too, that images containing text are fragmented but presented to the user as "complete". Moreover, very often spammers use html/css-based tricks to inject text into the content of the email, dirtying all the analysis indicators carried out on the text by automatic systems, but avoiding that this is read by the recipient (e.g. text of the same background color, text with "display: none" option etc.). Emails present a deceptive appearance: while the images and logos are copied perfectly, sometimes the only clues that are available to the user are the tone of the language, spelling errors, or other signs of unprofessional design.

Psychological Vectors. Attackers not only use technical vectors but also psychological ones, which exploit social engineering techniques. Psychological vectors aim to convince the victim to click on a link or open a provided attachment so that the attacker can, for example, collect personal information. These vectors exploit the cognitive vulnerabilities most present in people (e.g obedience to authority figures, gratitude etc.), applying actual principles of persuasion. Chapter 5 discusses these aspects in detail.

Combining the different techniques, in different specific contexts and with different end goals, yield different types of phishing attacks, which are discussed in the next section.

1.4 Types of Phishing email attacks

Emails, as explained in the previous sections, are the historical vector for phishing campaigns, and due to the growing awareness of users of illegitimate communications sent through email, criminals have started to exploit also new attack methodologies. Unfortunately, by exploiting public information about victims (e.g. posted on social networks), their employment status, and new email features and uses, phishing attacks are no longer just generic but also very much targeted to the victims. The robustness of people to these new attacks is even much lower than to non-targeted phishing attacks. Thus, different types of e-mail phishing attacks can be categorised, which differ mainly in method and target:

- ***Spear Phishing***: phishing attack targeted toward a company or individual; by referring to pre-acquired business or personal information, it is possible to compose a message that is very misleading to the recipient, who is inclined to trust it and carry out the action requested by the malicious sender. Names of colleagues or family members, company logos, topics of interest, are all information that aims to lure the victim.

- ***Lateral Phishing***: in a lateral phishing attack, adversaries leverage a compromised enterprise account to send phishing emails to other users of the company, benefiting from both the implicit trust and the information in the hijacked user account [36, 37]; once even a single company account is compromised, it is very easy to compromise others due to lateral phishing. This is one of the reasons why the problem of phishing is much more acute in large companies and it is very important to defend all users against phishing attacks.

- ***Clone Phishing***: it requires the attacker to create an almost identical replica of a legitimate message to trick the victim into believing it is real. The email is sent from an address that looks like the legitimate sender, and the body of the message is identical to that of a previous message. The only difference is that the attachment or link in the message has been swapped with a malicious one.

- ***Whaling***: an attack aimed at prominent figures within a company or a country's political scene. The goal is to manipulate the victim into divulging information in their possession or having them perform specific actions that are harmful to the company but profitable to the attacker.

The news is full of CEOs fired because they were defrauded by phishing scams, authorising transactions worth millions of euros.

- ***Calendar Phishing***: phishing links are delivered via calendar invitations. When calendar invitations are sent, they are automatically added to many calendars by default.

To further enhance the effectiveness of email phishing attacks, criminals sometimes combine them with fake SMS communications (*Smishing*) or even voice calls (*Vishing)*.

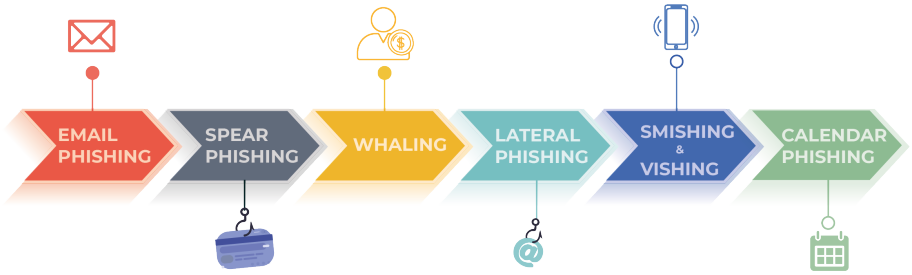


Figure 1.2: Types of email phishing attacks

1.5 Phishing prevention and detection

Before going on to the related work section, where we discuss the latest scientific findings aimed at solving the phishing problem, we summarise in this section the phishing defense methods that are most commonly applied today (Figure 1.3). However, it being still a hot topic of research suggests that they have not solved the problem satisfactorily.

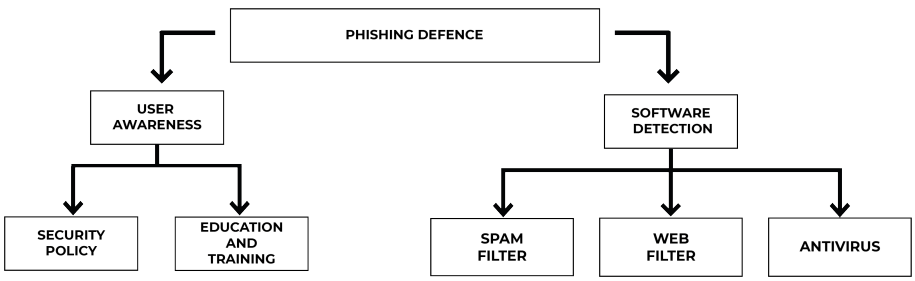


Figure 1.3: An overview of phishing prevention approaches

The solutions deployed to prevent security incidents caused by phishing emails must try to secure both systems, using classic IT security methods, and users' behaviour. To strengthen the security of the systems (e.g workstations of employees), three main security tools are used:

- **Spam filters**: when an email arrives, different logics are applied to detect an unwanted email upstream, before delivering it to the recipient. This is traditionally done with blocklisting approaches, based on the source IP addresses and the network-level behaviour of the emails (SMTP headers). More innovative detection logics, however, are also based on the content of the email, using pre-trained classifiers with machine learning technologies or simple heuristics. Unfortunately, criminals very often succeed in evading the logic of spam filters.

- **Web filters**: users are prevented from browsing websites known to be malicious. In this way, even if a user clicks on a malicious link contained in an email, browsing is blocked. Unfortunately, domain and URL reputation systems need to be continuously updated, and very often criminals create new (unknown) websites specifically for the purpose of carrying out phishing attacks. New branches of computer security are evolving to quickly share this information (e.g. Threat Intelligence).

- **Antivirus software**: users are prevented from executing malicious attachments. They are pre-scanned by antivirus software and classified as malicious or not. As with Web filters, it is important to keep malware signature databases up to date or to have intelligent malware detection logics. Unfortunately, malware often uses obfuscation techniques that prevent antivirus software from detecting it.

Given the non-full security coverage of the systems described above, and the increasing leverage that attackers give to the human factor, defense methods aimed at improving the user's security posture become very important. The most commonly used are:

- **Security policy**: 'simple' rules for proper use of email communication tools. Users are asked not to spread their email address too widely, not to use it for non-work purposes (e.g. subscribing to services or websites that could be subject to data leakage) and not to disseminate any work-related information on public channels. As far as possible, personal information on the net should also be reduced, as we know it is

used for social engineering techniques. Enhanced security controls are also often suggested for emails from outside the company perimeter (but internal emails also need to be checked because of lateral phishing).

- **Education and training**: security departments of companies (or third parties) often conduct regular education and training campaigns to make users aware of phishing, what to look for and how to prevent attacks. The way these courses are designed is crucial for their effectiveness, and is still the subject of scientific studies. If not well done, they might even be counterproductive. The work in this thesis also confirms that it is important for users to be trained to report to the security department emails that they have received and recognised as malicious (or that are suspicious).

This section concludes the background on the methodologies and types of attack and defense concerning phishing. In the next section, we describe the results obtained so far by the scientific community in solving this very pressing problem, from which the need for the contributions made in this thesis work in the following chapters becomes evident.

Chapter 2

Literature Review

The phishing problem has experienced an enormous increase in published papers and publications in the scientific landscape over the last 15 years. Although many efforts have been made towards the discovery of a reliable, permanent solution, the intrinsic complexity of this matter makes such task still an open problem, since no current solution comes without its shortcomings. This is the case, for example, of list-based approaches that make use of *Open Source INTelligence (OSINT)* information, such as lists of suspicious / dangerous DNS domains, that prove ineffective against *zero-day* attacks, which are not so uncommon in this domain. Several studies have tried to explain why phishing is so effective, focusing mainly on the characteristics of the attack medium and therefore on technical parameters; however, in order to achieve the goal of this thesis, described in the introduction section, it is necessary to perform a thorough analysis that studies the characteristics of humans and e-mails and correlates them, to understand which e-mails are effective against which users is needed. As a matter of fact, according to *Allodi et al.* (2019) [11] the main issue when dealing with phishing lies in the fact that technology is often unable to capture the human dimension, which plays a fundamental role in attacks based on *Social Engineering*. This is mainly due to the lack of a clear formalization of the vulnerabilities, characteristics, and processes of *Social Engineering* attacks, which could represent the foundation to devise more effective countermeasures.

In the early years of discussion of the phishing problem, most efforts were made towards the discovery of a technical solution. As time passed, the focus began to shift to solutions that took into account the psychology involved. Users' susceptibility with respect to phishing is claimed to be closely tied to

their cognitive processing of the e-mail. Unlike face-to-face communication, which provides both parties with rich background information and body language cues, e-mail communication is very flat and lacks information. This makes e-mails less persuasive, but also less suspicious [38, 39].

The state of the art on technical solution to spam emails and mail-based phishing attacks will be now explored; afterwards, the related works that focused on the human factor of phishing will be discussed.

2.1 Technical Solutions to Spam e-mails

Regarding *spam e-mails*, many studies have focused on finding countermeasures to the various techniques used by spammers, which resulted in a noticeable mitigation of this problem in present days, where most anti-spam filters are very effective against the vast majority of spam attempts. For obvious reasons, remedies to this kind of problems are purely based on a technical solution, since the psychology of the recipient is not involved. *Ramachandran et al.* (2006) [40] proposed a spam filter technique based on the analysis of the network-level behavior of spam e-mails. The idea was that spammers can vary significantly the content of e-mails, but they can't easily vary the way packets behave on the network, so it should be easier to detect spam by analysing how e-mails behave at a network level. By collecting Internet packets related to both malicious and legitimate e-mails, they were able to study the differences and identify the behavior of malicious e-mails and found that:

- most spam is being sent from a few regions of the IP address space (mostly from the Asiatic continent);
- that spammers appear to be using transient 'bots' that send a limited number of e-mails in a short span of time;
- that spammers take advantage of short-lived BGP routes to obstruct tracing operations.

According to *Van Der Toorn et al.* (2018) [41] about 15% of the spam belong to the category of *snowshoe spam*: a framework used by spammers to avoid detection by spam reputation systems (blocklists) which operates by sending spam e-mails originating from a large number of hosts. These attack strategies are also usually paired with practices to increase credibility, such as the correct configuration of the *Sender Policy Framework* (SPF) to ensure authentication of the sender domain; however, such configurations for a large number of hosts

result in the definition of many DNS records associated with the same domain; to this end, the authors proposed a system to detect snowshoe spam by means of *active DNS measuring* and Machine Learning techniques.

Al-Duwairi et al. (2013) [42] propose an image e-mail spam filtering technique that focuses on low-level image features to nullify the effect of obfuscation techniques usually employed in image-based spam e-mails. Specifically, by extracting low level image features such as Image Gradient, Run-Length Matrix, Co-occurrence Matrix, Auto-regressive Model and Wavelet Transform, they were able to use these features as input for different Machine Learning algorithms, hence finding the best one at predicting whether the image is spam or legitimate.

2.2 Technical solutions to Phishing e-mails

Among the most widely used technical tricks to convince the victim of the legitimacy of the received email, there are the following:

- inclusion of an official logo in the body of the e-mail;
- spoofing of the sender domain (when possible);
- writing style and text length mimicking;

which constitute a baseline for the building of any mail-based phishing attack. However, there are also more specific tricks that attackers usually implement. To give an example, a commonly used technique to evade anti-spam filter engines is such as the insertion of high quantities of **interference text** using *HTML/CSS*-based escaping tricks. Countermeasures to this kind of strategies involve the analysis of the **visible text** through an *Optical Character Recognition* (OCR) tool and the computation of the differences with the *HTML* content.

According to *Blanzieri and Bryl* (2008) [43], the main problems in classifying spam e-mails are:

- finding a labeled data set to perform supervised learning;
- the tradeoff between false positives and false negatives;
- facing new anti-spam filters evasion techniques.

To provide a solution to the first problem, this thesis also gives particular attention to the generation of a reliably labeled data set to be used as a ground truth

for the training of Machine Learning algorithms. Attempts to find a solution to the first problem also involved the use of semi-supervised Machine Learning algorithms such as the aforementioned RBF SVM [44] [45]. As for the second problem, according to *Michelakis et al.* (2004) [46], it is a good strategy to let clients choose which type of classification algorithm they prefer; for example, if a high false positive rate is not a problem, algorithms with higher recall and lower precision are the best option. Regarding the third problem, also highlighted by *Dada et al.* [47], as far as Machine Learning is concerned, there is a need to constantly find new features as a baseline for the classification of spam e-mails. In this regard, Gansterer and Pölz [48] in 2009 proposed six new features that can be extracted with ease, also when the e-mail flow is high, as well as ten more features with higher extraction costs. *Basavaraju et al.* [49] in 2010 proposed a method, based exclusively on the analysis of the text of the e-mail, to evaluate the importance of each word and construct a Term Frequency - Inverse Document Frequency (TF-IDF) model, which works particularly well with Deep Learning approaches. *Hamid et al.* (2014) [50] developed a system that makes use of clustering techniques to detect phishing e-mails; specifically, after extracting the 10 most important categorical, continuous and mixed data features, these were used as input for the clustering algorithm that was able to generate a profile of the analyzed e-mail and predict the class label: phishing or legitimate. With this thesis we have also expanded the number of features that could be extracted from the e-mail, also finding which are the best ones at predicting the danger associated with them.
Although being the most used model, supervised learning is sometimes replaced by models based on Deep Learning (DL) and Natural Language Processing (NLP). *Alhogail et al.* (2021) [51] propose a phishing email classifier model that applies Deep Learning algorithms using a Graph Convolutional Network (GCN) and NLP algorithms to improve the accuracy of already existing techniques for phishing detection.

In these approaches, the model leverages the information provided by the NLP algorithm, which often proves to be particularly efficient in extracting useful information from a text, to use it as input data for the Deep Learning algorithm. Such models have grown enormously in recent years because they can obtain high classification accuracy with reduced processing times, making them the best option for real-time classification tasks. However, NLP can also be of practical use when paired with supervised approaches. *Bountakas et al.* (2021) [52] propose a model for the classification of spam emails based on the combination of a NLP algorithm and a ML algorithm. To improve the perfor-

mance of ML classifiers in terms of accuracy and training time, they used the output of three different NLP algorithms, used to extract textual features from the e-mail, as input for five different ML algorithms with the goal of finding the best combination of NLP/ML algorithms in terms of *recall*, *accuracy*, *precision* and *F1-score*. *Word2Vec* proved to be the best NLP Algorithm, while *Random Forest* proved to be the best ML algorithm when the dataset was balanced and *Logistic Regression* proved to be the best one with an imbalanced dataset.
The phishing problem is most certainly facilitated by the way the Simple Mail Transfer Protocol (SMTP) was designed. SMTP was invented in a time where, of all concerns, security was the last. For example, SMTP defines a native method called ***VRFY*** [53] which allows to perform an enumeration of the e-mail addresses registered on a certain domain, hence facilitating the execution of spam and phishing campaigns.

2.2.1 SMTP Vulnerabilities

The SMTP-based e-mail system was not designed for the very diverse types of usage and application domains of today. For this reason, SMTP has been the subject of many scientific works and a large part of them focused on mechanisms to make the protocol secure. That is the case of Holst-Christensen and Frøkjær (2021) [54] who highlighted how, since SMTP servers were designed as open relays, allowing all clients to send via any SMTP server, the development of the very first security extensions of SMTP systems regarded sender authentication. To grant authentication, SMTP servers can make use of three protocols:

- ***SPF***, RFC 7208 [55], which defines a DNS record that declares which hosts are, and are not, authorized to use a domain name for the "HELO" and "MAIL FROM" identities. It advertises the list of mail servers that are allowed to deliver emails from a certain domain;

- ***DKIM***, RFC 6376 [56], which is an assertion that separates the question of the identity of the signer of the message from the purported author of the message. It does so by allowing an entity to sign a certain domain and claim responsibility for it by using techniques based on asymmetric cryptography.

- ***DMARC***, RFC 7489 [57], which is a mechanism that advertises information to instruct the receiving mail server on how to react on messages

where the SPF or DKIM information does not match or where the message header information conflicts with the SPF or DKIM information.

According to *Holst-Christensen et al.*, even large organizations have problems configuring and/or maintaining their email system in a secure manner; their analysis focuses on how the security extensions were implemented, as well as the mistakes and wrong assumptions made during their implementation. For example, many e-mail servers might run an old, vulnerable version of *Transport Layer Security* (TLS) to protect communications, thus exposing the communication channel to known attacks (such as **Heartbleed** or **Logjam**); furthermore, e-mails are stored in clear text, so host security is fundamental to avoid loss of confidentiality. Generally speaking, for each security extension to function as expected, it must be implemented by both the sending and receiving mail servers, which is often not the case. As a consequence, the more extensions a mail server implements, the more potential conflicts must be handled when configuring these extensions. Intuitively, the mail server with the lowest security level acts as a security bottleneck in all communications involving it. Therefore, the authors believe that instead of facing continued struggles with the basic shortcomings of the SMTP-based email communication framework, it might be more feasible, or even necessary, to investigate what can be achieved through alternative designs of fundamentally new email systems, based on protocols in which security, privacy and simplicity are the key concerns.
Riabov et al. (2005) [58] published a survey paper collecting all the most important details about SMTP, with a section dedicated to security issues. For example, it is mentioned that, according to *Campbell et al.* (2003) [59], most SMTP-specific vulnerabilities occur from misapplied or unapplied patches related to *Sendmail* installations or misconfigured *Sendmail daemons* on the SMTP servers.
We, therefore, believe that technical solution, such as *anti-spam* and *anti-phishing* filters, can be useful in certain situations, but do not represent a sufficient countermeasure to the evolving phenomenon of phishing that needs to be addressed in a more specific way, given the danger associated with such attacks.

2.3 Human Factor

Phishing e-mails are known to exploit the weakest link in cyberspace: the human. The psychology of the recipient of a phishing attack is a determining

factor in the success of the attack. *The human factor* has been the subject of many studies on phishing, for many reasons.
According to *Stajano et al.* (2011) [60]: *"a wise security designer would seek a robust solution which acknowledges the existence of these vulnerabilities as unavoidable consequence of human nature and actively build countermeasures that prevent this exploitation"*. For this reason, this thesis will focus on analyzing the effects that the exerting of persuasion principles used by phishers have on the target of phishing attacks.
Van Der Heijden et al. (2019) [10] prove the existence of a correlation between certain cognitive vulnerabilities and the effectiveness of phishing attacks, providing a quantitative estimate of this correlation. Although this result appears to depend on the specific application domain, its existence is really important, as it lays the foundation for the psychological analysis described in this work.
According to *Parrish et al.* (2009) [61] the ***Big Five*** personality traits, namely *openness, conscientiousness, extraversion, agreeableness, and neuroticism*, are commonly used in phishing studies and their correlation with phishing susceptibility is assessed.
Dhamija et al. (2006) [62] have been among the precursors in the field of phishing security and the first to prove that many users fail to distinguish between fraudulent URLs and their legitimate counterpart (e.g., they might think www.ebay-members-security.com belongs to www.ebay.com). In their study, *Dhamija et al.* asked candidates to speak about their thought process while explaining the reasons why they consider a content legitimate or fraudulent. Their results suggest that age, sex, education level, and frequent use of a computer are not good predictors of the detection ability of phishing, meaning that there is no significant correlation between these variables. On the other hand, knowledge about computer science, HTTPS, and certificates has proven effective in increasing the chance of distinguishing between legitimate and fraudulent content. An important result is that, despite a user's knowledge in relation to websites, HTTPS, certificates, and so on, phishing can still catch a user out of attention and be effective at deceiving it, for example, by placing two 'v's instead of a 'w' in the URL. For this reason, in our work we also take into account how many hours the user has worked before taking the test, as we believe that tiredness can play a determining role.
Some of these results are confirmed by one of the largest studies on phishing, conducted by *Lain et al.* (2021) [63]. Specifically, they have conducted an analysis on over 14'000 employees of a large company, employed in different

job roles, using realistic e-mails and in a time span of 18 months. Results show that: employees under the age of 20 are very susceptible, whereas employees in the 20-29 and 60+ year old categories are less susceptible; gender is not a significant predictor; putting warnings on top of suspicious e-mails is effective at preventing victims from falling for the phishing; detailed warnings are not more effective than generic ones; redirecting victims that clicked on a malicious link to a training web page surprisingly increases their susceptibility to phishing compared to people who did not receive such training, despite the training material was designed by a specialised company according to best practices and guidelines defined in scientific literature.
The efficacy of warning messages showed to users that click on a malicious link is assessed by *Kumaraguru et al.* (2009) [64]. They conducted a study on 515 participants and demonstrated that delivering a training message when the user clicks on the URL in a simulated malicious e-mail decreases their chance of clicking on a malicious link after two days and the efficacy lasts for over 28 days. They have also demonstrated that this kind of warnings do not decrease users' willingness to click on links in legitimate communications. In this study, users were asked to insert some information about them and the authors found that gender is not a factor in predicting phishing susceptibility while, with respect to age, users in the 18-25 years old category appear significantly more vulnerable than older users.

Due to the importance of the psychological factor, phishing emails are often constructed by including what is known as *cognitive attacks*: portions of sentences used to exert a certain *persuasion principle* on the victim. Persuasion principles were first introduced by Cialdini in "*The psychology of persuasion*" in 1984 [13] and refer to psychological tricks properly crafted to exploit specific *cognitive vulnerabilities* of the victim. These cognitive vulnerabilities are expressed as follows:

- **Authority**: tendency to obey people in authoritative positions, driven by the possibility of punishment for not complying with authoritative requests;
- **Liking**: tendency to say "yes" to requests from people the individual knows and likes. People are "programmed" to like others who show appreciation for them and who are similar to them;
- **Scarcity**: tendency to assign more value to items and opportunities when their availability is limited, not to waste the opportunity, and not to feel

regret;

- **Consistency**: tendency to behave in a way consistent with past decisions and behavior. After committing to a certain view, product or action, people will act in accordance with those commitments;
- **Social proof**: propensity to label behavior as correct to the degree of others performing it;
- **Reciprocity**: desire to repay others when a favor is received.

A cognitive attack is the insertion, inside the e-mail, of specific sentences used to exert a certain principle of persuasion on the recipient with the purpose of exploiting one or more of its cognitive vulnerabilities. These sentences can be included in a generic way or in a more precise way. The difference between these two practices is the location, inside the e-mail, where the cognitive attack is placed. For example, a sentence that expresses a sense of authority can be placed in the body of the e-mail or can be placed in the footer of the e-mail with a notation called *contact information*, and have different success rates depending on the adopted method. Notification methods describe a set of practices that can be employed to increase the effectiveness of the attack.
According to *Burda et al.* (2020) [65] certain *notification methods* are best suited to deliver specific cognitive attacks. Specifically:

- **Contact Information - Authority**;
- **Personalization - Liking**;
- **Subject Line - Scarcity**;
- **Subject Line - Consistency**.

where **contact information** is a notification method which indicates the inclusion of a name, an e-mail address, a business address, a phone number, typically added at the bottom of an e-mail; **Personalization** is a notification method which consists in personalizing the e-mail content to the identity of the recipient, their location, and preferred language to increase the feeling that the message is custom made for the recipient; **Subject Line** is a notification method which involves the inclusion of a semantically meaningful and captivating subject line to capture the attention of the recipient and thus help increasing e-mail opening rates.

The authors also found interesting results in regard to tailored-phishing attacks, such as high values of correlation between the inclusion of the *authority* psychological attacks in the email and the exposure to phishing, which will be more thoroughly discussed in the next section. Keeping in mind that **tailored phishing** and **large-scale phishing** have completely different effects on the psychology of the recipient, the effectiveness of the psychological vector *authority* in large-scale phishing is assessed by *Quinkert et al.* (2021) [31] who analyzed the impact of 14 psychological and technical attack vectors of more than 400,000 e-mails sent to employees of 77 different companies and found a positive correlation between the use of the authority psychological vector and the number of clicks on malicious links. This work studied the effectiveness of the approach adopted by PhishCo: a company whose aim is to increase the awareness regarding large-scale phishing of the employees of the client company. With this approach, the employees of a client company receive a large number of fake phishing e-mails over a long period of time and the employees who click on a phishing link are notified that the e-mail was sent by PhishCo. This training methodology proves to increase the awareness of the employees of the company by reducing the number of clicks on malicious links over time. PhishCo client's employees receive fake phishing e-mails continuously for a long period of time, keeping the attention threshold high and avoiding running into the limitations about the duration of the effectiveness of common training campaigns.

2.4 Tailored Phishing

Tailored phishing is a class of attack where e-mails are crafted by including believable *Social Engineering* artifacts, providing a high level of sophistication. Such attacks are different from the most common ones meant to target as many people as possible.
Burda et al. (2020) [65] conducted an experiment involving employees from a university (UNI) and from a consultancy company (IND), classified by their role inside the organization: Junior, Senior, Support. Their goal was to study the impact of professional role on the success rate of tailored phishing attacks, as well as the impact of notification methods and whether cognitive attacks are effective. Their results suggest that: junior employees are more vulnerable than senior and support ones, especially in the UNI; in presence of a tailored attack, the overall persuasiveness of a well-engineered e-mail is a bigger factor than the cognitive vulnerabilities of the recipient; the delivery of the Authority

cognitive attack more than doubles the chances of success of phishing attacks directed at the UNI employees, while halves the success rate when aimed at employees of the IND, showing that employing techniques to deliver cognitive attack can amplify the effectiveness of the attack or decrease it based on the recipient of the e-mail. Finally, they show that despite receiving phishing awareness training, e-mails sent to IND employees have higher success rates than emails sent to UNI employees, which is a result that confirms the limitations of the currently employed training practices.
From these considerations, it is clear that user awareness is the key in the phishing scenario.
Pirocca et al. (2020) [66] describe how to create targeted attack simulations that can be used for training and awareness campaigns in an organization, incorporating all the procedures that attackers have to follow.
Targeted phishing campaigns are aimed at increasing the likelihood of the attack success by creating customized e-mails and landing pages that vary based on the characteristics of the victim.
To achieve this, attackers need to go through certain phases, comprising:

- **Open Source INTelligence (OSINT) gathering**: information gathering on the victim through *Social Engineering* techniques;
- **Sending Profile creation (domain selection)**: creation of a trust-able sender domain, preferably from the internal domain of the company the victim is working in;
- **Landing Page cloning**: cloning the website the victim is lured to by modifying the page based on the needs;
- **E-mail Template creation**: creation of the e-mail used as a vector artifact for the attack by using the information gathered during the first phase;

As mentioned, according to RFC 7489 [57], an e-mail domain (e.g. @ebay.com) can be spoofed if no DMARC policy is set for it. If the DMARC policy associated with a mail server is set to 'discard', all e-mails with the said spoofed domain will be discarded.
For this reason, the selection of the domain for the sending profile is carried out by analyzing the DMARC policy associated with an e-mail server with a certain domain and verifying if no policy has been configured.
When sender domain spoofing is not possible, attackers must resort to the use of domain squatting techniques to mimic the original domain, explained in

Chapter 1.3. With regard to the awareness campaign, the customization of the e-mail is obtained by inserting some keywords inside the text of the e-mail that can take on different values depending on whether certain conditions are met. Conditions refer to the characteristics of the target, such as nationality and topics to which they are sensitive. In this way, based on the information collected about the target, the body of the e-mail can be adapted to increase the chances of the attack being successful. For example, if the user is sensitive to climate change, a token placed in the text could verify the characteristics of this user and place a sentence on the climate change topic to catch his attention.

2.5 Evaluation of the Scientific Approach to the Phishing Problem

To give the reader a visual representation of the contributions of the scientific works that have been mentioned, Table 2.1 highlights how, as time progressed, the focus shifted from technical analysis to psychological ones. Specifically, works on the rows are sorted in chronological order. When a work introduces a contribution that never appeared before, a column is added; in this way, also the contributions are sorted in chronological order. When two or more works share the same exact typology of contribution, they are placed in the same row. Analyzing the columns of the table it is possible to see how the early works introduced more technical solution to phishing such as security at SMTP level, while the latter works introduced more human-oriented contributions, such as studies on tailored phishing and frameworks to increase the current awareness campaigns effectiveness. It is possible to see a trend which highlights how, as time passed, human-oriented contributions have become the focus and that all the latest works capture the human dimension.

In contrast with existing literature, this thesis introduces a framework that focuses on the several possible types of email attacks, incorporating multiple phishing countermeasures [67] coming from different domains (e.g. human aspects, URL blocklisting, protocol analysis etc.). Such a framework is used to guide and optimise the work of anti-phishing analysts in enterprise settings. The existing studies, instead, present countermeasures coming from a single domain or focus on a specific type of attack. Our models are trained and evaluated on the information that a security analyst generally has at his disposal when analysing a security incident generated by an email, available when the email is reported as unsolicited. Therefore, the input to our models is not a

generic email but the reporting of an unsolicited email. This is a significant aspect because of the inherent difficulty in distinguishing an unwanted email from an actual cyber attack. The novelty lies in this triage task and in the estimating of the probability of a phishing attack to succeed, considering both its technical aspects towards the systems and cognitive aspects towards the victims. Our approach uses traditional supervised machine learning algorithms, but with novel objectives and novel input information. Despite the strengths of supervised learning, it is often impossible to apply due to the absence of a reliable labelled dataset [68]. With the big effort of users and analysts of the company, an extensive and reliable ground truth has been collected in two years. This dataset has been used to build automatic classifiers and to achieve new contributions and significant results about the effectiveness of the various existing countermeasures, the characteristics of successful phishing attacks, and the cognitive vulnerabilities of humans about phishing.

After working on this field for a few years, it has become clear that there is an urge to understand in detail the cognitive aspect of the phishing phenomenon, to design practical and effective solutions. In the second part of this thesis we introduce for the first time in literature a system to collect data for this purpose. Such a system analyzes a variety of backgrounds of users, such as work background, education background, computer skills, and personality. Moreover, this analysis is conducted creating e-mails characterized by many diverse attack strategies observed in the wild in our previous (and also documented in literature), from the spoofing of the sender domain, to the various domain squatting techniques, the inclusion of images, the employment of cognitive attacks. Such system is used to provide useful material for research on the phishing problem, totally public and available to the scientific community, but also to spread knowledge about what dealing with phishing e-mails means. The existing studies, instead, propose framework for the analysis of just specific groups of users or use e-mails characterized by reduced diversity in terms of their characteristics. The novelty lies in this added variety and in the evaluation of when and how certain users are more vulnerable than others or that certain typologies of e-mail are more effective than others, hence taking in consideration both cognitive/human and technical aspects. The information gathered this way can result crucial for the design of new awareness programs or tools that take in consideration different vulnerabilities of users.

	Analyse security at SMTP level	Study the effectiveness of semi-supervised algorithms	Study the effects of domain squatting	Captures the Human Dimension	Analyses network-related behavior	Expand the set of features for ML models training purposes	Study the effects of tailored phishing	Study the effectiveness of cognitive attacks	Define a framework for awareness campaigns
Campbell (2003) [59] *Riabov* (2005) [58] *Stringhini* (2012) [69] *Holst-Christensen* (2021, survey) [54]	✓								
Chan (2004 [45]) *Dai* (2019) [44]		✓							
Dhamija (2006) [62]			✓	✓					
Ramachandran (2006) [40] *Van Der Toorn* (2018) [41]					✓				
Gansterer (2009) [48] *Basavaraju* (2010) [49]						✓			
Parrish (2009) [61]				✓					
Nikiforakis (2014) [35] *Agten* (2015) [32] *Kintis* (2017) [33] *Quinkert* (2019) [34]			✓						
Cidon (2019) [70]				✓			✓		
Van Der Heijden (2019) [10]				✓				✓	
Pirocca (2020) [66]				✓			✓		✓
Burda (2020) [65]				✓			✓	✓	
Lain (2021) [63]				✓					
Quinkert (2021) [31]				✓				✓	✓

Table 2.1: Related work overview. The topic of the research work is shifting more and more from the technical aspects to the human factor of phishing

Chapter 3

Collaborative Framework against Phishing

This section presents the "life cycle" of a spam email in the partner company. In the scenario of this work, the defense against attempts at spam fraud follows a collaborative approach: in addition to commercial email filtering systems, there is a system of collection of reports that allows the computer emergency response team (as defined in [9]) to protect the affected users thanks to the recognition of a threat by a more aware/expert user. This distributed approach is important for detecting security incidents that would otherwise go unnoticed. It is just as important, as a prevention strategy, to conduct periodic awareness campaigns to train users to recognize email-based cyber attacks, in order to increase their awareness of risk and to educate them in reporting such attempts to breach the company's security. As explained in [71], in fact, user training plays an important role in reducing their vulnerability to phishing. *Burda et al.* [72], on the other hand, explain how the users' reports constitute an important, and sometimes underestimated, resource that can provide an indicator of the dangerousness of a suspicious email, thus helping to speed up response and mitigation actions. Moreover, according to *Burda et al.* [72], the most experienced users able to identify the attempts of fraud by email, rarely report the email to the appropriate departments, thus denying the analysts an important help to detect an eventual security incident.

Typically, when a security incident occurs, one or more of the following recovery actions are undertaken, in increasing order of relevance:

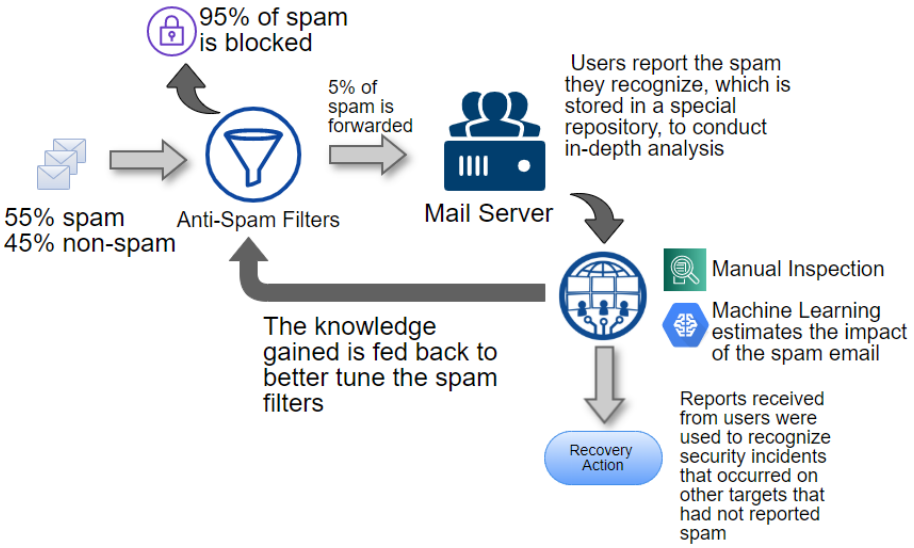


Figure 3.1: Ecosystem of spam defense

- Sending notification to all users involved about malicious email detection;
- Adding filters in the navigation proxies to block navigation or downloading from malicious or otherwise unknown sources;
- Rehabilitating of nodes and networks compromised by any malware. Resetting of accounts and credentials that may have been violated;
- Technically analyzing in-depth attachments and links, in order to get a thorough understanding of their risk and adequately protect affected users;
- Investigating on perpetrators and taking possible legal actions.

The purpose of our collaborative framework is both to recognize and resolve security incidents that have occurred, and to intercept them before they occur. The prediction of which spam emails will actually generate a security incident can leverage machine learning techniques. Estimating and assigning an accurate level of risk to each reported email is extremely important. The number of incoming reports is huge. Prioritizing their analysis allows security experts to deepen the investigation exclusively on the most relevant ones. The

ecosystem illustrated in Figure 3.1 has allowed to collect over time the spam messages that reached users and to memorize which of them has led the recipients to download an attachment or to browse a link. This information is recorded directly in the tracking logs of the company navigation proxy, and can be requested by analysts only in the case of a clear possibility of a security incident. The analysis of these messages let us acquire a deep knowledge of the main features of most critical spam emails. In principle, the estimation of the risk score of the email could be made upstream for all emails with unsupervised approaches, even before a user reports it. However the process should not be time consuming, given the huge amount of emails to analyze. For this reason, the impact of feature reduction on classification performance has also been studied (Section 4.3).

In summary, as a result of our proposals, the collaborative defense approach relies not only on collaboration between users, who report suspicious emails also for the defense of others, but also on the collaboration between human and machine. In fact, thanks to the joint effort of employees and security analysts, a machine learning engine is fed with samples reported by users and labelled by analysts. The machine is then trained on what the main characteristics of the dangerous emails are, providing a classification on new reported suspicious emails to the analysts, and also provide important information on where users need to improve to avoid being victims of phishing attacks. When fully operational, the two phases become concurrent, forming a virtuous circle of defense and prevention (Figure 3.2).

3.1 Data collection and ground truth construction

Our data collection system was started in early 2018. Since then, whenever an employee receives an unwanted email and decides to report it to the security department, it is stored in our archive. All emails in the dataset are by definition spam emails. A large amount of additional security-relevant information about each element of the email is automatically computed and stored together with it. This is the typical information that the SOCs of all companies are supplied with (e.g. reputation rates of links and attachments from blocklists and antivirus systems, hierarchical level in the company of the recipient etc. see Table 3.3), in order to correctly manage this type of attack. For this reason the dataset is highly specialized, with information coming directly from the field and promptly made available to analysts. Very often such information is available only through the purchase of third party services such as reputation

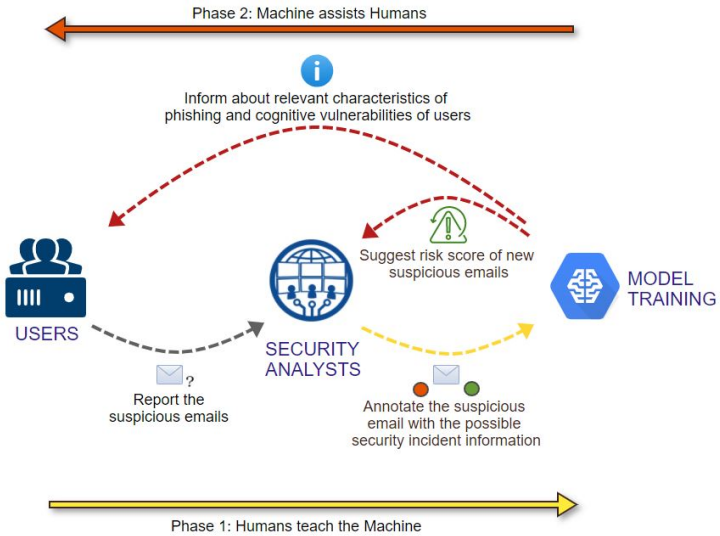


Figure 3.2: Humans and Machine collaboration framework

services, sandboxes, threat intelligence feeds, blocklisting services, etc. Based on this information, a specific group of security analysts composing the anti-phishing group, day by day checks if a security incident is generated by these incoming spam emails. Due to the enormous amount of reports that arrive every day, it is not feasible to perform a thorough security check for each of them, mainly because most of them represent simple noise. This first triage is very important, because it allows an initial filtering that would prioritize spam emails that need to be checked immediately; however, this distinction task cannot be delegated to the simple recipients of the email because it would require a strong security expertise to carry it out. The security check mentioned above is an extensive series of checks, such as the assessment of how widespread this email is in the mailboxes of all the other users, if someone has clicked on a malicious link, if he has downloaded a malicious attachment, if some workstation has been infected, if credentials have been violated, etc. Further on, these checks can also include the log analysis of navigation proxy, user agents installed on workstations, and sometimes also interviews to the recipients of the email. Finally, all the evidence found by analysts is also stored together with the spam report: whether the security incident was detected by analysts or not, and the (possible) remediation actions taken by analysts. This allows a manual classification and labeling of data performed by analysts as a result

of their daily work with the data collection system. The email reports in the dataset are categorized in two possible classes:

- **Critical spam - Label 1, Positive**: spam emails that have created a security incident or at least required a defensive action to prevent future infections;
- **Not relevant spam - Label 0, Negative**: spam emails with low or no degree of danger, and did not require any recovery action.

Formally, a dataset $\{(x_1, y_1), ..., (x_n, y_n)\}$ has been built where each sample x_i is characterized by a vector of m feature values $< f_0, ..., f_{m-1} >$ and has an associated class $y_i \in \{0, 1\}$. Several hundreds of reports from our 100,000 users are received every day. Many of these are duplications, because these types of attacks are often executed in large campaigns that target many recipients with the same email. Excluding the duplicates and not considering the many reports not processed due to their huge number, the dataset contains a total of 21,932 distinct samples: 3,931 were labeled as Critical/Positive, 18,001 as Not Relevant/Negative. This dataset is one of the major contributions of this scientific work, designed and constructed in the context of this doctoral work. The work of the security analysts, which was used to generate the labelling, was organized with several mutual consultation sessions. The dataset has been used to perform supervised machine learning and obtain a classifier that allows immediate recognition of the threats contained in the mails.

3.1.1 Annotation consistency evaluation

The scarcity of labeled datasets is a known problem in cyber security contexts, amplified by the difficulty that even a human may have in manually labeling a dataset. In other contexts, such as the recognition of a dog or a cat in a photo, for example, the labeling task is much simpler compared to the security analyst labeling job, which requires strong prior knowledge. Despite only hard critical and healthcare environment require an almost perfect level of reliability, we believe that a thorough analysis of manual labeling reliability is always necessary to ensure it does not undermine the correctness of the experiments. Due to the nature of the classification problem set, in fact, the human verdict on the positivity or not of a sample may be ambiguous, or may differ between the various expert analysts involved in the manual labeling. This is why, before starting our studies, we decided to evaluate the annotation consistency and the

inter-rater reliability of our manual labeling. These are typical problems of manually labeled datasets, well addressed by M.L. McHugh [73].

The main metrics used to measure the consistency of labeling and the inter-rater reliability are: the percent of agreement and the Kappa statistic [73]. The first, more traditional one is calculated by the number of agreement observations divided by the total number of observations. Its key limitation is that it does not take into account the possibility that raters guessed on the labels; actually it is a remote possibility in our case, given the experience of the raters, but it is known that all humans can make mistakes. The Kappa statistic was designed to take into account the possibility of guessing, nevertheless it has other kinds of limitations. It is calculated through the following:

$$\kappa = \frac{Pr(a) - Pr(e)}{1 - Pr(e)} \tag{3.1}$$

where $Pr(a)$ is the probability of agreement and $Pr(e)$ is the chance probability of agreement (function of row and column marginals):

$$Pr(e) = \frac{(\frac{cm^1 xrm^1}{n}) + (\frac{cm^2 xrm^2}{n})}{n} \tag{3.2}$$

Both percent agreement and Kappa statistic have strengths and limitations, but in brief it is possible to assume that the former is an upper bound and the latter is a lower bound of the annotation consistency. To compute these two metrics, two analysts (the most and the less experienced ones) have been asked to work on the same subset of spam reports (composed of n = 263 elements) in completely separate sessions, in order to see if they agreed on labeling. The main point to evaluate is the very first look that is given to the report, when the analyst decides whether the email has the potential to create a security incident. The following steps to ascertain whether a security incident has occurred are guided by more standardised procedures, as defined for example by ISO/IEC 27001 standard [74], based on the evidence found without any discretionary approach.

The results of this experiment are shown in the Table 3.1. The second analyst, the less experienced one, considers more spam emails as dangerous compared to the more experienced one; he probably feels less confident and prefers to get false alarms instead of ignoring potential incidents. However, the two analysts are 94.67% in agreement on the responses concerning 263 observations. The Kappa statistic computed with (3.1), applying the (3.2), is 83.3%. Therefore, the labeling of our dataset can be considered *strongly*

		Analyst 1		Row Marginals	
		Positive	Negative		
Analyst 2	Positive	42	10	52	rm^1
	Negative	4	207	211	rm^2
Column Marginals		46	217	263	
		cm^1	cm^2		n

Table 3.1: Data for Kappa calculation

reliable and can be used to train machine learning models to be deployed in this operating environment.

3.2 Feature set design

Starting from the raw information automatically collected when a spam email is reported, the set of features to be extracted and used as input to learning models has been designed. The full set of features extracted from the samples is listed in Table 3.3 and comprises 79 features. The features are grouped by the nature of the information from which they are extracted or the reason why they were thought to be good at discriminating between the two classes.

Each group of features, referred as *feature field*, is described in depth in the following.

[5]Features calculated twice: first on the text extracted from the email content and then on the text extracted with an Optical Character Recognition from the email displayed. Regarding the latter, the feature name used in the thesis is the same followed by the "_clean_text" suffix. The field of these features is referred to as "content_view".

[6]Features calculated twice. The alternative version of the features take the "_d1" suffix in the name. Refer to Section 3.2.1 for further information.

Phishing words				
account	security	user	verify	service
valid	required	credentials	attention	request
suspended	company	bank	deposit	post
Scamming words				
$	€	£	customer	prize
donate	buy	pay	congratulation	death
please	response	dollar	looking	urgent
warning	win	offer	risk	money
transaction	sex	nude		

Table 3.2: words considered deceiving (for scam and phishing purposes)

1. **General.** General information, mostly extracted from the SMTP headers: if any smtp server is blocklisted, size of the mail, number of recipients etc, plus all those features that give us information about the email's origin and destination.
   Rationale: These features are not expected to be very discriminating on their own, but they might be in correlation with others. Moreover, the dangerousness evaluation of an email based on its origin and SMTP path is a typical analysis made by anti-spam filters, and it may be useful in our classification task as well.

2. **Content.** Features extracted from the text in the content of the email: language, number of words, number of deceiving words, number of disguised words, readability indexes, simplicity and correctness of the text etc. As for "deceiving words", previous studies [75] show that the words listed in Table 3.2 are those most used to capture the attention of the scammed target. It has been manually verified that this is also true in our dataset. In addition, "disguised word" refers to a word which has an edit distance of 1 from the name of the company, the names of its subsidiaries and the names of its main partners. All the **Content** features have been calculated also on the text extracted with an Optical Character Recognition tool, generating the **Content_View** features (as described in the next feature field).
   Rationale: The actual message carried by an email is the content, which is also one of the main elements to analyze in order to detect the presence of an attack and estimate its effectiveness. It is the main means used by attackers to satisfy the first condition we deem necessary for the attack to succeed: the recipient must be subjugated. These features may allow classifiers to distinguish emails that are immediately trashed by the recipients from those that induce a mistake to the attacker's advantage. For example, the search for disguised words is useful because very often addresses or domains similar to those normally used by the company are crafted to deceive employees.

3. **View.** Features extracted from the screenshot of the email as it is displayed to the recipient: height and width of the screenshot, number of images, amount of text within the content but not read by the recipient etc.
   Rationale: These features have been selected to include in our analysis also the cognitive visual perception that the recipient has when opening

the email. Moreover, very often spammers use HTML/CSS-based tricks to inject text into the content of the email, polluting all the analysis indicators carried out on the text by automatic systems, but avoiding that this is read by the recipient (e.g. text of the same background color, text with "display: none" option etc.). Several features have been extracted with an Optical Character Recognition (OCR) tool, with a twofold objective: to detect differences between text contained in the email and text actually displayed, as an indicator of malicious behavior, but also to calculate the content features on the text actually read by the recipient (generating the **Content_View** features). The extraction process of these features is described in detail in Section 3.2.1.

4. **Subject.** Features extracted from the subject of the email: number of words, number of characters, if there are non-ASCII characters, if the email is forwarded or answered.
   Rationale: The subject line is the first thing the recipient reads of an email, and it is known [76, 77] to have a great importance for the communicative effectiveness of the message carried. For this reason the subject line is also expected to have a great value on how much a recipient can be fooled into believing in a message depending on the characteristics of the subject.

5. **Links.** Features about the links in the email: number, number of link domains, information from URL analysis service, etc.
   Rationale: Links can be the carriers of malicious content of an email, they must be carefully analyzed to quantify how much the email meets the second condition we deem necessary for the attack to succeed: the payload of the attack must not be trivial. In this perspective it is very useful to rely on information from online link and domain reputation systems, typically available to the SOCs of companies. VirusTotal has been used for this analysis[1].

6. **Attachments.** Features about the email attachments: number, type, size, information from sandboxes and antivirus, etc.
   Rationale: As with links, attachments can be the carriers of the malicious content of an email, they must be carefully analyzed for the same reasons. The information coming from sandbox and antivirus systems

[1] VirusTotal is an online malware and URL analysis service `https://www.virustotal.com/gui/home/upload`

can help, especially taking into account the specific systems used by the company.

7. **Other.** Other types of information not in the previous fields: number of malicious entities known thanks to Threat Intelligence activities, role in the company of recipients, etc.
   Rationale: Other information closely related to the company also can contribute to the identification of emails more relevant than the others: the strategic importance of the role of recipient or reporter in the company is very useful in assessing the risk that would arise in case of compromise. For example: if a deceived employee answers an email with information about personal agenda or meetings, it may not be considered a security incident. In the case of a manager, because of the sensitivity of the information he/she is dealing with, it certainly is. In addition, information from the Threat Intelligence Platform (TIP), which is an internal platform managed by the company's security department that aims to collect and share IoCs, has been included in this field. [2]

The designed feature set, comprising legacy and novel features, includes information considered in previous works, but never extracted and used as ML features, as well as information available to SOCs of companies but never used for these purposes. This feature set aims to capture the characteristics of (almost) all the attack techniques that criminals generally use in phishing, which are documented in Chapters 1 and 2. We believe this set of features represents an important contribution to the field, as no feature engineering work on these issues has ever been presented before, to the best of our knowledge, combining both technical and cognitive aspects, with the ultimate aim of prioritising the security processing of phishing emails. This represents a substantial step towards achieving the automation of this process.

[2] Indicator of compromise (IoC): in computer forensics is an artifact (e.g. antiviral signatures, malicious domains or IP Addresses etc.) observed on a network or in an operating system that, with high confidence, indicates a computer intrusion [78]. In this context IoCs are antiviral signatures, malicious IP Addresses, MD5 hashes that uniquely identify a malicious file, URLs and/or domain names from which an attack has been carried or to which a malware connects once activated.

Field	Feature	Description
General	is_html	if it is an HTML mail
	n_smtp_blockist	the number of smtp servers traversed in the blocklists
	email_size	the size of the email
	n_recipients	the number of recipients
	n_hops	the number of SMTP hops
	is_IT	if the email comes from Italy
	is_EU	if the email comes from Europe
	is_NA	if the email comes from North America
	is_SA	if the email comes from South America
	is_RU	if the email comes from Russia
	is_AS	if the email comes from Asia
	is_AF	if the email comes from Africa
	is_OC	if the email comes from Oceania
Content[3]	language[5]	the language of the mail
	voc_rate[5]	the rate of words of the content in the vocabulary
	vdb_rate[5]	the rate of words of the content within the basic vocabulary
	vdb_agg_rate[5]	the rate of adjectives within the content
	vdb_v_rate[5]	the rate of verbs within the content
	vdb_s_rate[5]	the rate of nouns within the content
	vdb_art_rate[5]	the rate of articles within the content
	gulpease_index[5]	readability index (Italian - Gulpease index [79], English - Flesch formula [80])
	n_words_content[5]	number of words in the content
	n_disguisy[5]	number of disguised words in the entire email (content, subject, address)
	n_phishy[5]	number of deceiving words, related to phishing, in the content and subject
	n_scammy[5]	number of deceiving words, related to scamming, in the content and subject
View	screenshot_width	the width of the email as it is displayed to the recipient
	screenshot_heigth	the heigth of the email as it is displayed to the recipient
	n_images	number of images
	n_images_links	number of images as links
	hidden_text[6]	percentage of text in the content not displayed to the recipient
	hidden_text_words[6]	number of words in the content not displayed to the recipient
	hidden_text_chars[6]	number of characters in the content not displayed to the recipient
Subject	n_words_subject	number of words in the subject
	n_char_subject	number of characters in the subject
	is_non_ASCII_subject	if the object contains non-ASCII characters
	is_re_fwd_subject	if the email is replied or forwarded
Links	n_links	number of links
	n_domains	number of link domains
	vt_l_rate	rate of links considered malicious by at least one engine of VirusTotal
	vt_l_maximum	maximum number of VirusTotal engines that consider a link as malicious
	vt_l_positives	number of links considered malicious by at least one engine of VirusTotal
	vt_l_clean	number of links not considered malicious by all engines VirusTotal
	vt_l_unknown	number of unknown links to VirusTotal
Attachments	n_attachments	number of attachments
	n_image_attachments	number of image type attachments
	n_application_attachments	number of application type attachments
	n_message_attachments	number of message type attachments
	n_text_attachments	number of text type attachments
	n_video_attachments	number of video type attachments
	attachments_size	average size of attachments
	vt_a_rate	rate of attachments considered malicious by at least one engine of VirusTotal
	vt_a_maximum	maximum number of VirusTotal engines that consider an attachment as malicious
	vt_a_positives	number of attachments considered malicious by at least one engine of VirusTotal
	vt_a_clean	number of attachments not considered malicious by all VirusTotal engines
	vt_a_vulnerable	number of attachments considered malicious by VirusTotal engines not including corporate antivirus
	vt_a_partial	number of attachments considered partially malicious by VirusTotal engines not including corporate antivirus
	vt_a_protected	number of attachments considered malicious by VirusTotal engines including corporate antivirus
	vt_a_unknown	number of unknown attachments to VirusTotal
Other	n_tip	number of entities in TIP
	n_tip_a	number of attachments in TIP
	n_tip_l	number of links in TIP
	n_vips	the number of vips among the recipients
	n_medium_vips	the number of managers among the recipients
	n_high_vips	the number of top managers among the recipients

Table 3.3: Features extracted from the raw data

3.2.1 Clean text extraction with optical character recognition

The process of extracting the clean text from the samples of our dataset is quite complex and consists of several phases, represented in Figure 3.3.

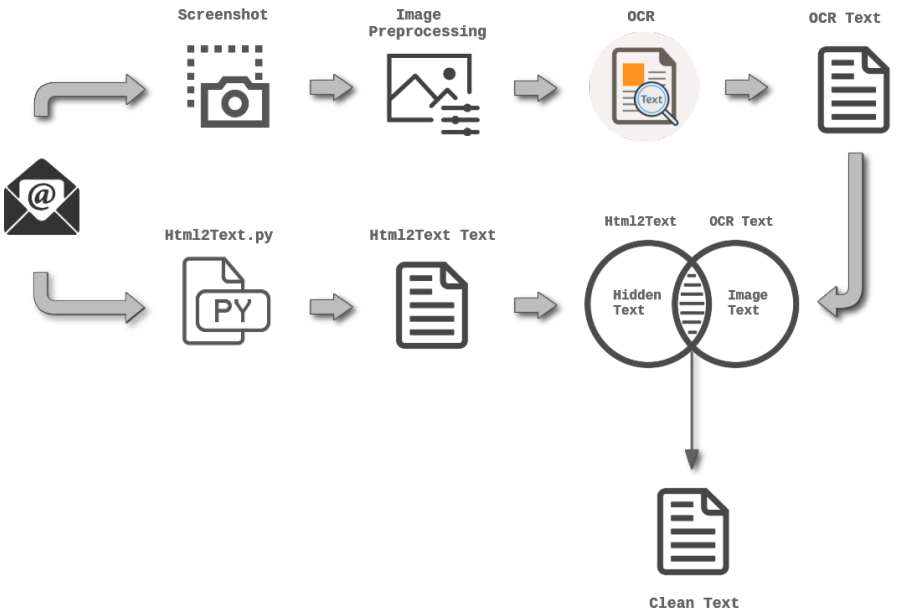


Figure 3.3: Clean text extraction scheme

The clean text is the result of the intersection of two text files that can be extracted from a single email: the text obtained by running OCR (Optical Character Recognition) on the screenshot of the email, which we call “OCR Text”; and the text obtained converting the HTML version of the email into a clean and easy to read text, which we call “HTML2Text Text”. The reason why we intersect these two text files lies in the main drawback of HTML2Text Text: being derived from the HTML of the mail message, in addition to the text that is shown to the user by the mail client used, it also includes all the text injected into the mail as Hidden Text. This can take the form of text with the same color of the background, for example white text on white background, but it can also be some text not shown by setting the property “display:none” or several other HTML/CSS-based tricks. The OCR Text, on the other hand, is the text obtained through OCR performed on the screenshot of the email rendered in the browser, i.e. all the text that the user can actually see and read

when he opens the email. This text, however, also includes all the text of any images in the email, i.e. Image Text. It is evident that the text obtained from the intersection of these sets, HTML2Text Text and OCR Text, i.e. the text in both of them, is precisely the clean text we are looking for. The OCR tool, however, can make some mistakes when recognising words, usually by misreading one character in the word. In order to handle this behavior, we decided to consider equal words with an edit distance of 1, thus generating two types of hidden text features: "hidden_text" and "hidden_text_d1". The "clean_text" version of the features are all obtained by using the "d1" version of the hidden text.

The problem therefore shifts to deriving the two main ingredients for the creation of the clean text. The HTML2Text Text can be simply obtained by using of the homonymous Python script in order to clean up the HTML code of the email from the various language tags. The operations necessary to obtain the OCR Text are much more complex and require a more detailed analysis and explanation. The whole process of OCR Text extraction, as shown in Figure 3.3, consists of three main steps, all automated through Python scripts:

- Rendering the email in the browser and saving the screenshot
- Post-processing the screenshot
- Text recognition in the screenshot by OCR

The first phase was carried out using Selenium, a set of tools designed to automate browsers, and, more specifically, by using Selenium Webdriver. Through an appropriate Python API you can access all the features offered by Selenium Webdriver in a simple and effective way. In order to work, Selenium needs an appropriate driver to communicate with the chosen browser interface. This driver, which varies from browser to browser, must be downloaded and installed before you can run any Python code related to Selenium. In our case, having chosen Chrome as the browser to automate, the driver is made available by Google and is called "chromedriver". Selenium Webdriver allows you to manipulate DOM elements in Web pages and to control the browser through appropriate Python commands. It is possible for example to start a new browser instance, make it open the email and capture a screenshot of the screen. In order for all these procedures to work correctly, the webdriver was configured to start the browser in *headless mode* with an opportune window size, a zoom of 450% and with a timeout limit of 2 seconds. This configuration was required to be able to open a web page not limited in size by the

display in use and to capture the email, full loaded, with a single high resolution screenshot. The OCR tool used, Python-tesseract, in fact, requires images with a recommended resolution of at least 300 DPI.

As we know, optical character recognition systems are born and are designed to detect the characters contained in a document, i.e. essentially black writings on a white/yellow background. Emails, on the other hand, in addition to having text on a white background, can assume the most varied shapes and colors. This makes the screenshots we have acquired unsuitable to be processed by an OCR tool as they are. It was therefore necessary to edit the images in order to ease the recognition of the characters. The solution we found after countless tests, consists in converting the screenshot into grayscale, in order to reduce the chromatic variability, and applying a sharpening filter, to make the text stand out more from the background.

As a last step, the screenshot thus obtained and modified, has been processed through the OCR tool Python-tesseract, a wrapper from Google's Tesseract-OCR Engine. For a complete description of the tool and the various possible configurations, please refer to the project page [81]. The configuration setup involves recognizing Italian as the main language in the text, and English as a secondary language. The extracted text has been saved in an appropriate text file, and has been used together with HTML2Text Text to extract the Clean Text.

3.3 First look at the collected dataset

Before starting to use the data collected with Machine Learning techniques, this section shows a preliminary view on the data collected, so that initial analytical considerations can be made. It is important to know your data before feeding it to machine learning engines, which do not always provide feedback on their inner logics. The Figures 3.4, 3.5, 3.6 and 3.7 show how the feature values of the samples in the dataset are distributed, divided into the two classes *Critical Spam* and *Not relevant spam*, in order to have insights on which characteristics better highlight the dangerousness of a spam email. The heatmaps in the Figures show that in general, spam emails have only one recipient, indicating that attackers prefer to send the same email several times to individual targets, rather than to groups of multiple recipients. Critical spam differs from non relevant spam in having more hidden text, well-written and easily understandable content, and a relatively short and concise subject line. In addition, in potentially dangerous emails, malicious content is usually delivered through an

attachment or a single link to a remote repository. This factor, combined with the compactness of the email, indicated by a screen length of approximately 600 pixels, the general absence of images and an increased use of malicious terms as shown in the 3.4 table, contributes to catching the victim's attention to the intended target. It is no coincidence that the most frequently used words in the most effective phishing emails are aimed at exploiting people's cognitive vulnerabilities, as discussed in more detail in the Chapter 5 of this thesis.

Critical spam				
greetings	*payment*	*invoice*	*account*	*link*
cordial	~~*email*~~	~~*telecomitalia*~~	*kind*	*customer*
grant	*must*	*thanks*	*service*	*pay*
opportunity	~~*messagge*~~	*data*	*receipt*	*order*
by	*renewal*	*breakthrough*	*time*	*information*
attachment	*count*	*deadline*	*fast*	*now*
Not relevant spam				
season	*county*	*serie*	*own*	*part*
first	~~*email*~~	*house*	~~*telecomitalia*~~	*contract*
against	*championship*	*career*	*click*	*production*
second	*years*	*family*	*receive*	*life*
property	~~*messagge*~~	*presence*	*territory*	*company*

Table 3.4: Most present words in the emails of the dataset, excluding conjunctions and prepositions. Those common to both classes are crossed out. The respective translations have been inserted instead of words in the original language (Italian)

It was mentioned in Section 3.1 how the labelling of the samples collected in the dataset was carried out. Explaining the process in more detail, the emails that created a security incident need to be further characterized, for forensic purposes and to identify the best mitigation action in the specific case. During this enhanced characterization, the relevant email is specialized into one of the classes listed in Table 3.5, representing the different possible email attacks.

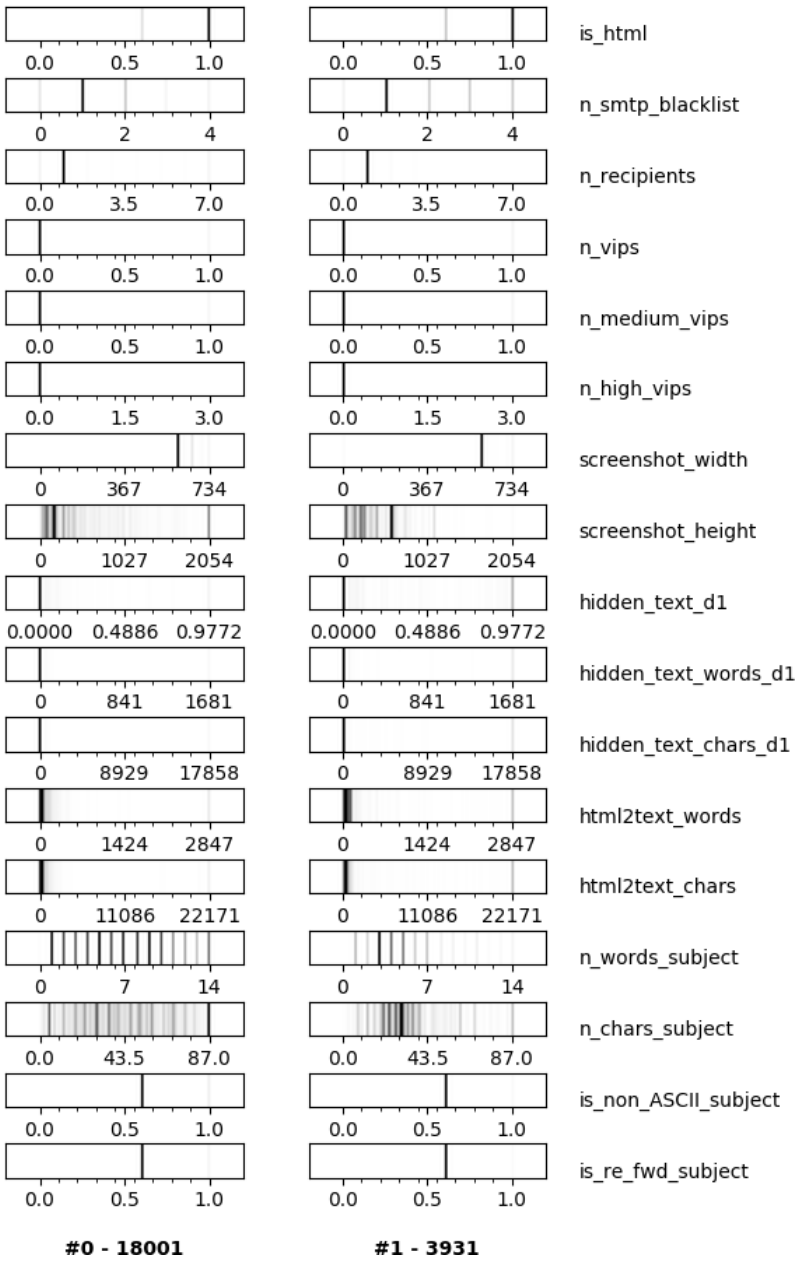


Figure 3.4: Heatmaps of the distributions of feature values (1)

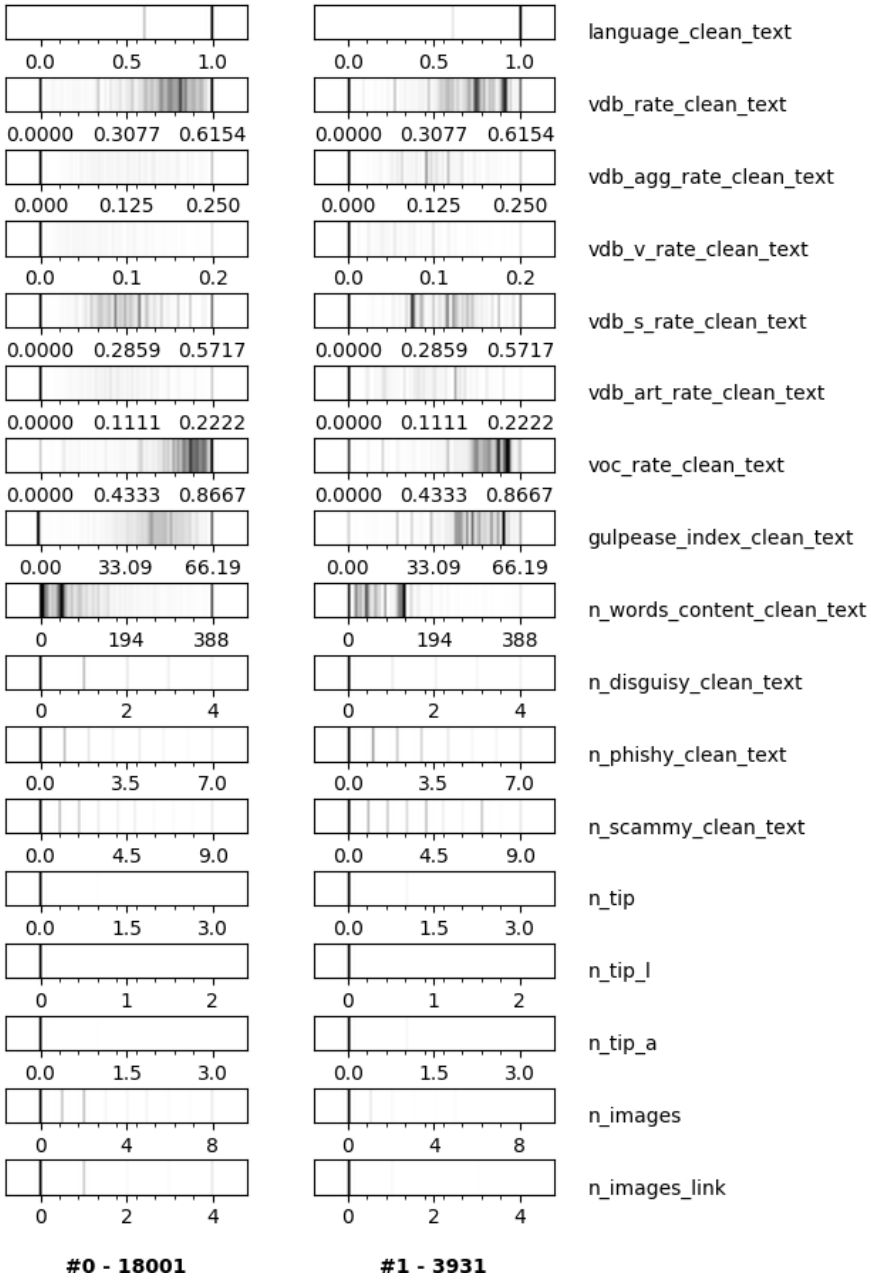


Figure 3.5: Heatmaps of the distributions of feature values (2)

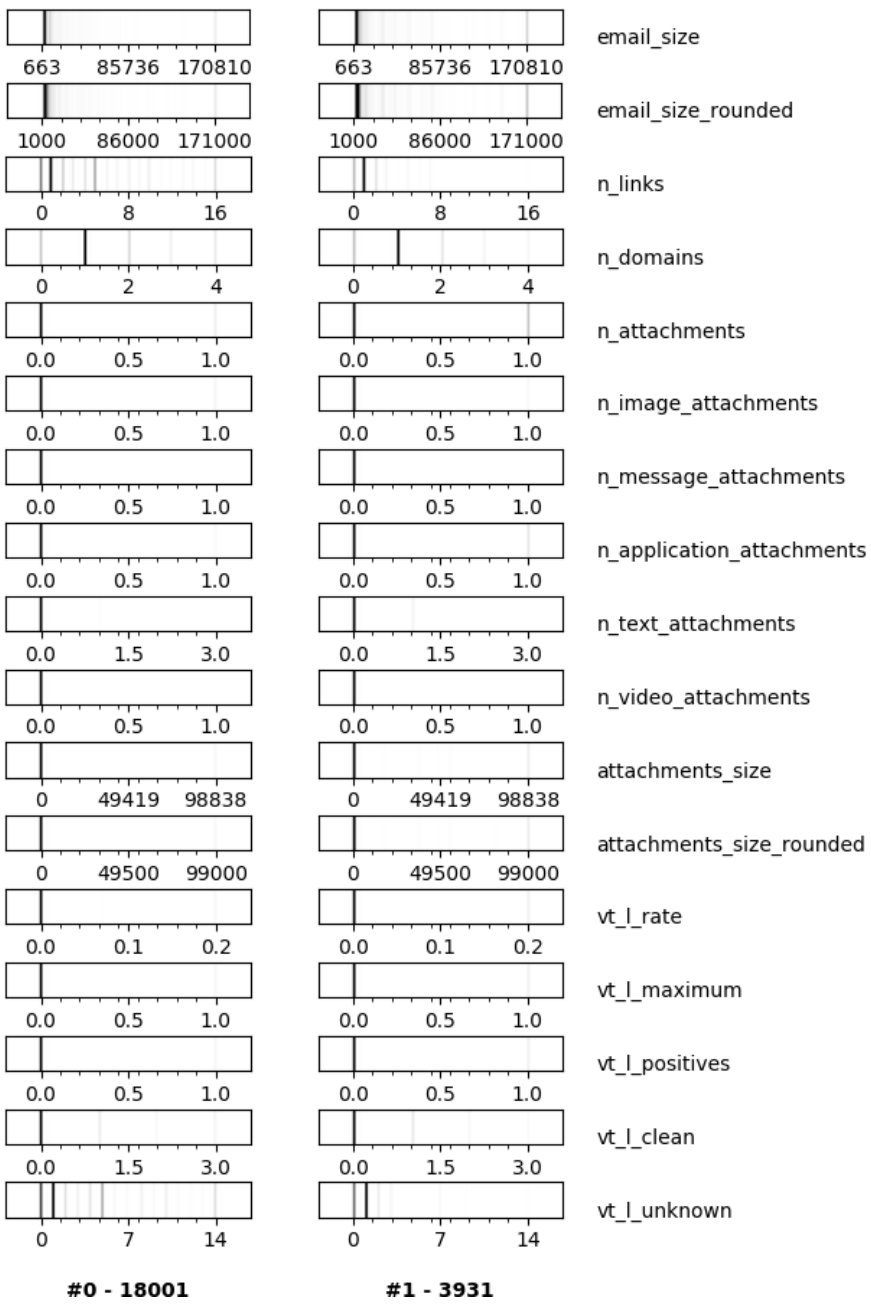


Figure 3.6: Heatmaps of the distributions of feature values (3)

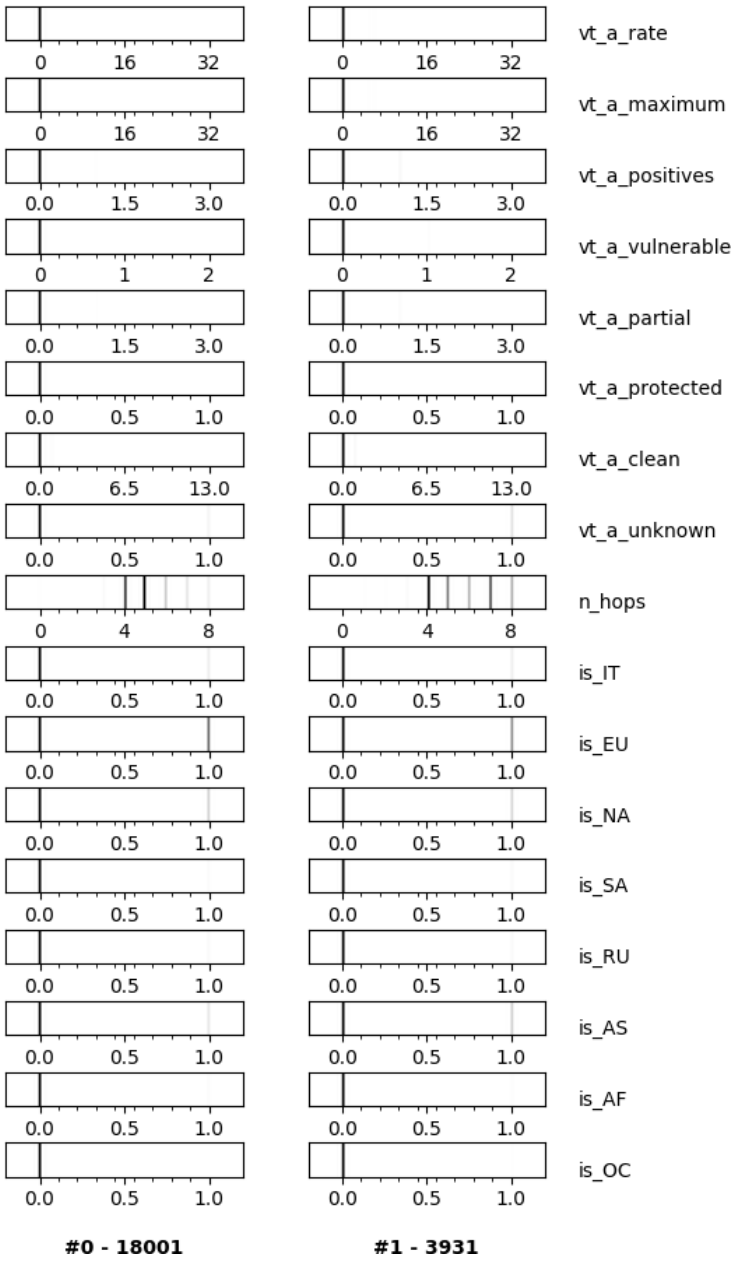


Figure 3.7: Heatmaps of the distributions of feature values (4)

Class	Description
Malware Propagation	E-mails conveying malicious content (e.g. Ransomware, Trojan) into the company through attachments, links and other channels
Ceo Fraud	E-mails that attempt to induce prominent persons within the company to perform actions dangerous to the company, such as authorising monetary transfers or sharing sensitive data; these are the messages that make the most extensive use of social engineering techniques
Phishing Enterprise	E-mails attempting to steal credentials of company accounts by means of phishing techniques
Trademark Abuse	E-mails containing references to web pages and other online content that misuse the corporate brand to carry out malicious actions
Scam	E-mails on company mailboxes containing attempts to defraud the recipient, such as the theft of credit card data
Phishing	E-mails attempting to steal credentials of generic accounts (not corporate accounts, but still belonging to the employees) by means of phishing techniques

Table 3.5: Specialized classification

The 3,931 positive samples in the dataset are divided into this deeper classification as shown in Figure 3.8. It is clear that the types of attacks, with their different inherent purposes, are very unbalanced across the 6 classes. The most frequent attacks are those carrying malware, followed by attempts to steal account credentials using phishing web pages. In smaller numbers, but potentially very dangerous, are CEO Frauds, for which only improvements in security awareness can represent an effective protection, as they often arrive without any attachments or links to analyse. Since each of these attack classes are handled by performing different actions (among those listed in Chapter 3), it might be interesting to design a multiclass classifier, which would not only highlight which of the e-mails needs to be checked by a security analyst (contribution of this thesis work), but also identify the necessary mitigation action from the outset. This would bring us closer to a more automatic handling of the whole process, ideally without the intervention of the human analyst. Unfortunately, according to our experiments, the performance of the multi-class classifiers tested with our feature set does not perform satisfactorily. The reason for this unsatisfactory performance is that the type of attack is inferred by analysing not only the email itself, but also the linked web pages, the behaviour

of malwares/attachments, and other in-depth analyses that are not done on the email alone, but mainly on the other 'artefacts' it carries. A broader analysis should therefore be designed beyond the email alone: this certainly represents one of the most interesting future work.

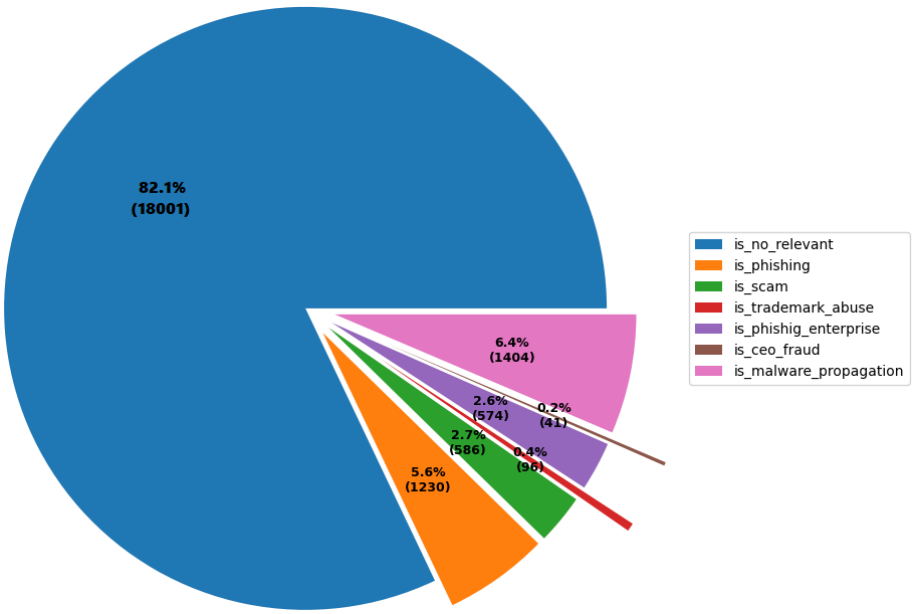


Figure 3.8: Distribution of samples in the different attack classes

Chapter 4

Experimental analysis and results

The available dataset has been divided into two parts: training set (85%) and test set (15%). The training set was used in the analyses related to the choice and optimization of model, its hyperparameters, weights, threshold, and features. These results have been validated through 10-fold cross-validation. The test set was used to evaluate the actual performance of the models properly tuned and optimized in the previous phase.

4.1 Selecting supervised Machine Learning models

Machine Learning models have been trained to perform the binary classification explained above, with the aim of choosing the best ones and conducting further in-depth experiments on them. Scikit-learn and the following ML-based algorithms have been used: Nearest Neighbors, Linear Support Vector Machine (Linear SVM), Radial Basis Function Support Vector Machine (RBF SVM), Decision Tree, Random Forest, Adaptive Boosting (AdaBoost), Naive Bayes, Quadratic Discriminant Analysis (QDA), Multi-layer Perceptron Neural Network (MLP Neural Net). These ML models have been selected on the basis of the experiments shown by other works concerning spam detection [43].

The classification capabilities of these nine supervised approaches have been tested by computing the True and False Positive Rates (TPR/FPR), using as input the full set of features. Figure 4.1 depicts the Receiver Operating Characteristic (ROC) curves obtained with each model. These results have

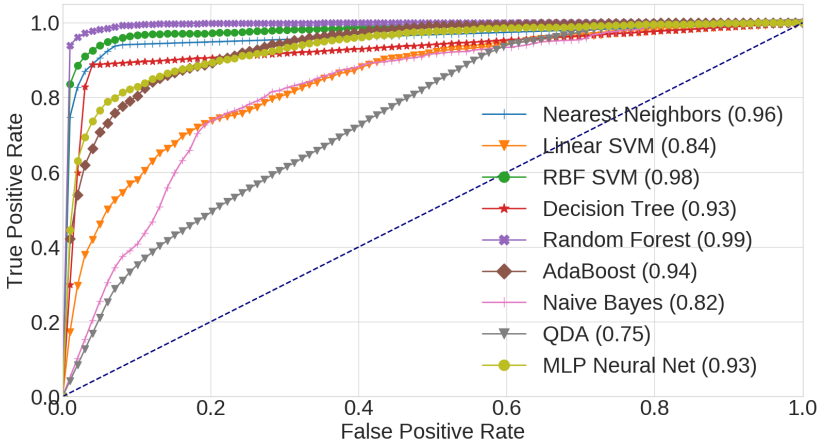


Figure 4.1: ROC curves of different ML-models (AUC values)

been obtained with a 10-fold cross validation on the training set. One of the metrics used to evaluate the performance of these approaches is the "Area under Curve (AUC)", which shows that the two best approaches are Random Forest (99%) and RBF SVM (98%). Random Forest has been configured with 140 trees in the forest and 8 variables in the random subset at each node, following the optimization process proposed by *Lee et al.* [82]; RBF SVM has been configured with a gamma coefficient of 0.7 and a penalty parameter C of 5. Since the dataset is unbalanced, the AUC alone cannot properly evaluate the performance [83]. For this reason it has been used only for a preliminary selection of the best models, whereas all the following results are shown in terms of Precision and Recall. The Precision and Recall metrics of the two best-performing approaches are evaluated in details in the following section as a function of the class weights, classification thresholds, and feature sets.

4.2 Tuning hyperparameters, class weights, and threshold value

This section explains the approach adopted to properly tune the two models previously selected: Random Forest and RBF SVM. The optimization on the training set of the hyperparameters of such models has been automated using the *RandomizedSearchCV* and *GridSearchCV* functions made available by Scikit-Learn. The functions specify the set of values to be tested for each hy-

perparameter. In the case of Grid Search, the system is evaluated on all combinations of values of all hyperparameters, while Randomized Search randomly draws values of hyperparameters from the specified distributions, performing a predetermined number of iterations. The best value of the hyperparameters was found by first using *RandomizedSearchCV* to identify the order of magnitude and reduce the range of values to be tested, and then *GridSearchCV* to fine tune the search of the optimal values. According to the tests performed, Random Forest achieved the maximum performance of 98.5% Precision and 89.1% Recall with the following hyperparameters:

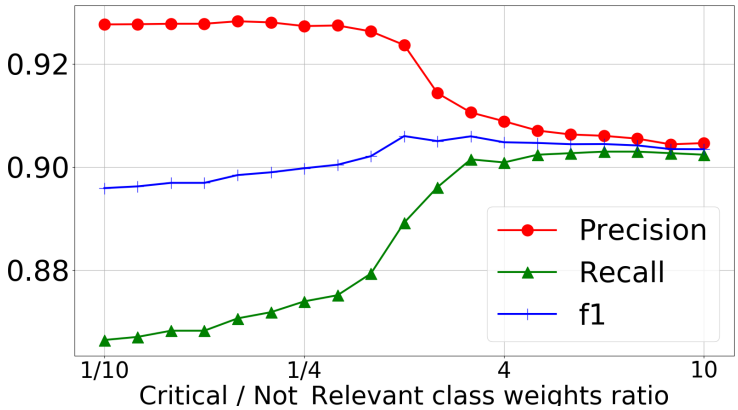
- **n_estimators**= 700
- **max_features**= 'auto'
- **max_depth**= None
- **min_samples_split**= 2
- **min_samples_leaf**= 1
- **bootstrap**= True

RBF SVM achieved the maximum performance of 92.4% Precision and 88.9% Recall with the following hyperparameters:

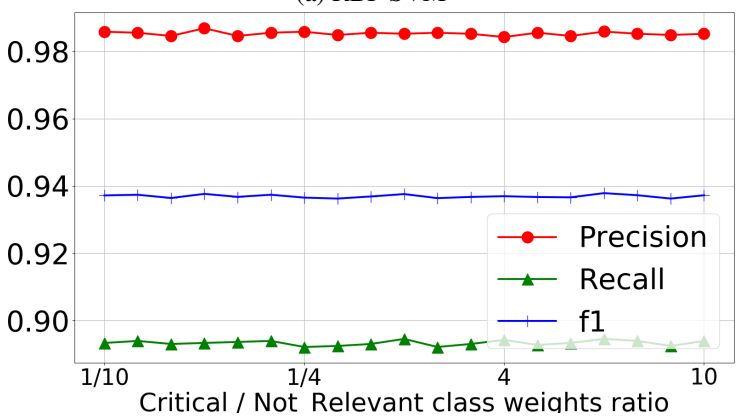
- **C** = 15
- **gamma** = 0.7

Subsequent analyses are then performed using these hyperparameter values for the two models.

Figure 4.2 shows the Precision, Recall and F-measure of RBF SVM and Random Forest, varying the weights assigned to the two classes. Random Forest has better performance in general (F-measure of 93.8%). On the other hand, RBF SVM obtains higher Recall values (90.3%) at the expense of the Precision (90.6%). In some contexts RBF SVM may be preferred to maximize Recall (up to 90.3%) and minimize risks. In other context, and probably more in general, the tradeoff with Precision (going down to 90.6%) means an excessive amount of false alarms. Random Forest has slightly smaller values of Recall (up to 89.1%), but much higher values of Precision (up to 98.5%). The class weights chosen are therefore:

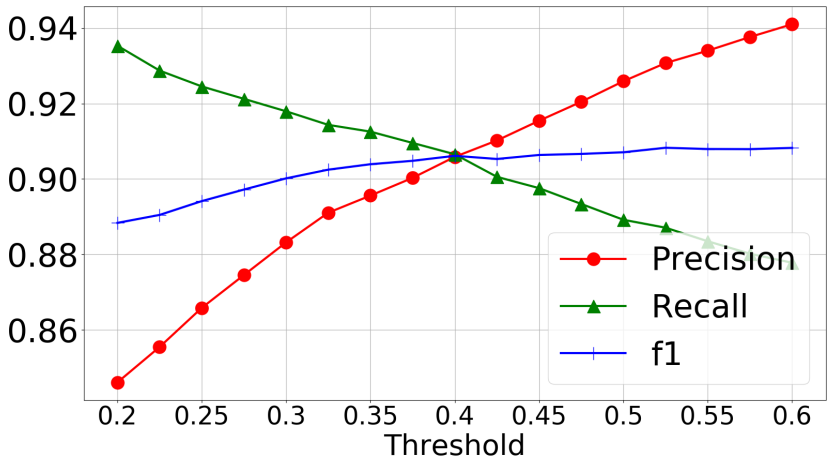


(a) RBF SVM

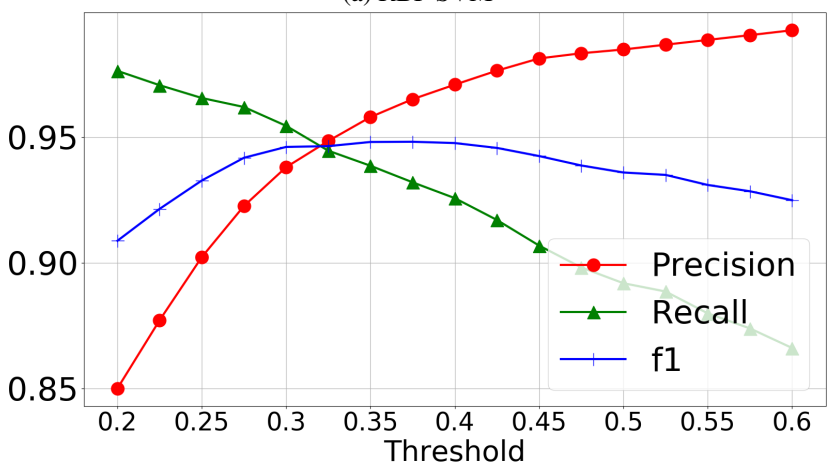


(b) Random Forest

Figure 4.2: Performance with different class weights



(a) RBF SVM



(b) Random Forest

Figure 4.3: Performance with different classification threshold values

- **Random Forest** Positive: 1, Negative: 1
- **RBF SVM** Positive: 3, Negative: 1

The classification threshold has been tuned using the class weight reported above. Figure 4.3 shows the Precision, Recall and F-measure of RBF SVM and Random Forest as a function of the classification threshold value. According to Figure 4.3, Random Forest shows better performance than RBF SVM even in contexts where the Recall is more important. Analyzing the graphs at peak of the F-Measure curve, Random Forest Recall (93%) achieves higher values than RBF SVM Recall (88.8%), but keeping much higher Precision values (96.5% vs 93%). In addition, even the best RBF SVM Recall value achievable (93.5%) barely beats the previous Random Forest Recall. Random Forest is therefore the best supervised Machine Learning model for our purposes. The best classification threshold values are 0.525 for RBF SVM and 0.375 for Random Forest.

4.3 Feature Ranking

The importance of each feature is analyzed in this section. Two types of analysis have been performed: a first one that considers the individual contribution of each feature, and a second one that considers the contribution of each feature as part of the full feature set, therefore considering the correlations among each other. The first analysis allows to deepen the cognitive phenomenon at the basis of phishing attacks, highlighting the features that have a significant impact in making some spam emails critical compared to others. These results are a fundamental guide to conduct awareness campaigns for the mail recipients, to train them to handle the specific cognitive vulnerabilities they have shown. The second analysis is more concerned with technical aspects: evaluating the real informative contribution of each feature in the context of all the others may lead to identify a subset of features bringing optimal classification performance. Using fewer features however reduces the complexity of the processing to be performed, the execution times and the costs. Moreover, a large number of features does not always correspond to an improvement in performance, due to redundant information, noise in the data and overfitting.

To estimate the individual predictive power of each feature f_i, the mutual information between it and the discrete (binary) target variable y has been computed. The results are shown in Figure 4.4 and show that the distinguishing characteristics of successful email attacks mainly concern the way in which

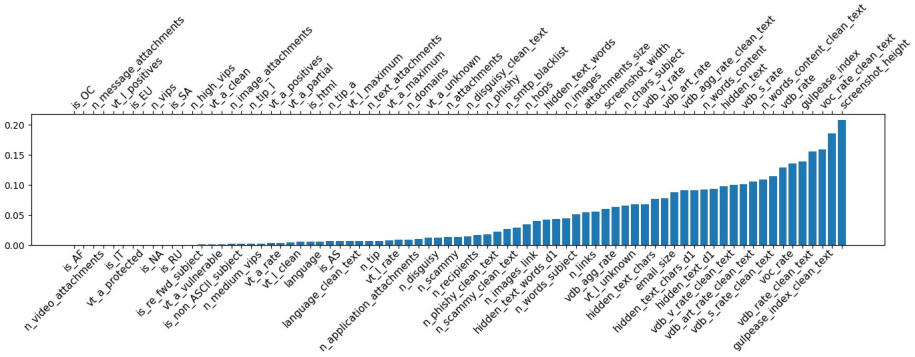
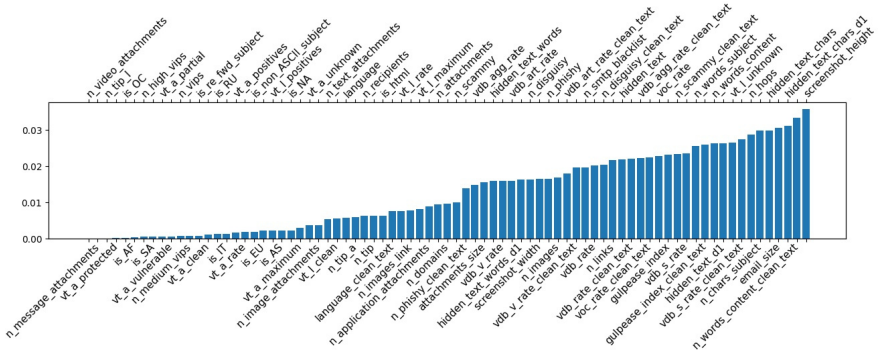


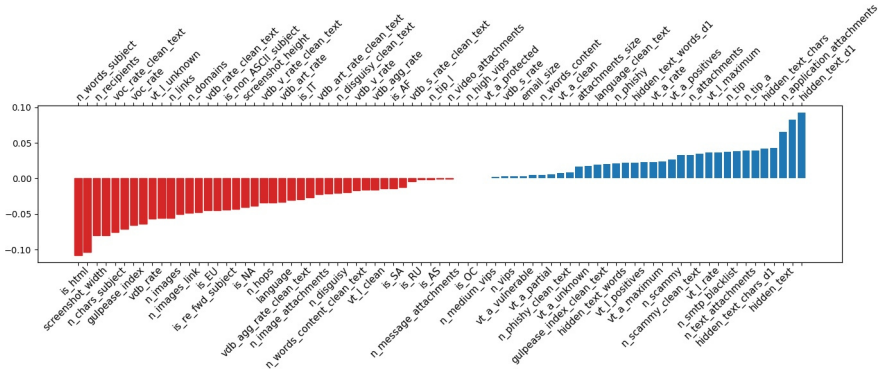
Figure 4.4: Mutual Information between features and positive class: how much information the feature contributes to the classification

the content is written. The indexes that estimate the readability of the text evaluating the punctuation, how the message is dispersed in height, the degree of correctness and simplicity of the syntax, and terms used are all very relevant characteristics. Figure 4.4 also shows that the "Content" features in the "clean text" version, are almost all more important than the "normal" ones. This confirms the need of a method to identify and isolate the hidden text injected in the emails. Interestingly, the origin country of spam is not a discriminating factor to identify critical spam, while features related to reputation systems such as VirusTotal and SMTP blocklists, as well as the number of SMTP hops, provide a quite important contribution.

The Wrapper methodology [84] has been used in order to select the best subset of features: it consists in using the prediction performance of a given ML model (named Wrapper) to assess the relative usefulness of subsets of features. In the case of Random Forest the importance of a feature represents how much that feature has contributed to decrease *Gini's impurity*, and this can be easily calculated. As for RBF SVM, instead, computing the actual importance of a feature is a complex procedure as also confirmed by *Liu et al.* [85]. For this reason, SVM with a linear kernel has been used to compute the feature importance. The results of these studies are shown in Figure 4.5. They confirm what has already been discovered thanks to mutual information analysis, with some interesting additions: the number of recipients and words of the subject are also relevant for the classification. Moreover, information deriving from threat intelligence processes also acquire importance if related with the other features.

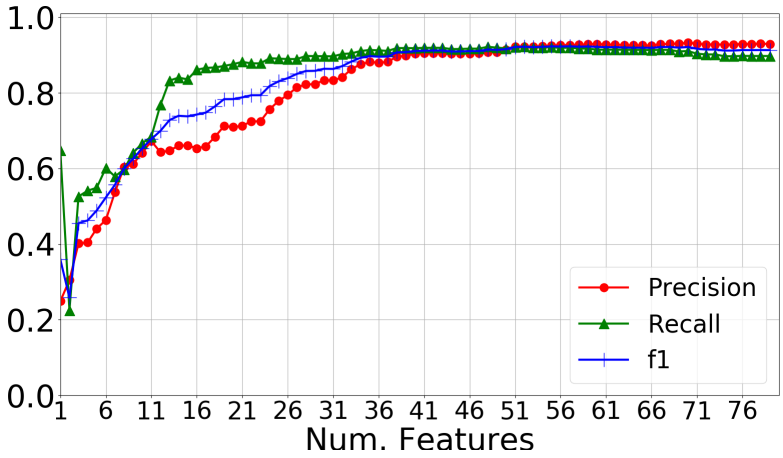


(a) Random Forest as Wrapper

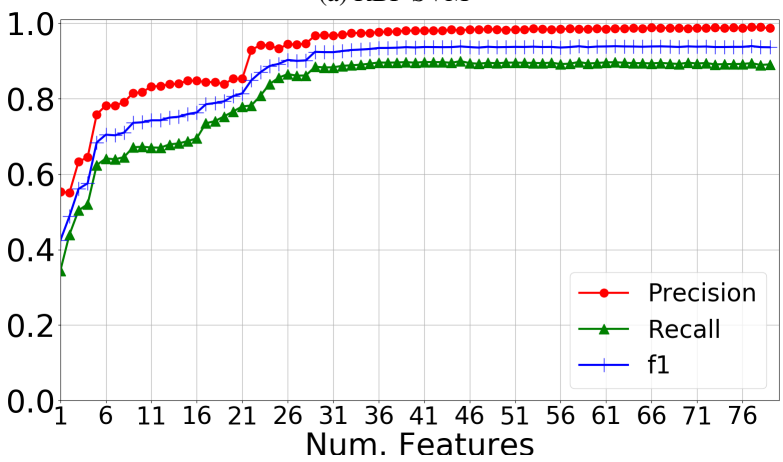


(b) Linear SVM as Wrapper

Figure 4.5: Feature importance with Wrapper method



(a) RBF SVM



(b) Random Forest

Figure 4.6: Performance with increasing number of features

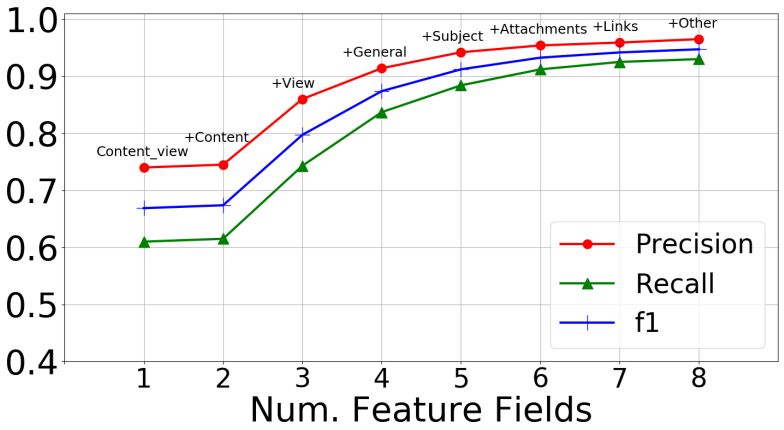


Figure 4.7: Random Forest performance with increasing number of feature fields

The impact of the number of features on the classification performance has been evaluated. The Recursive Feature Elimination procedure has been used for this aim. The results are shown in Figure 4.6. In both cases, using the entire feature set is counterproductive: the performance of the classifiers slightly degrades while training and classification times increase. The best performances for Random Forest and SVM are achieved with 36 and 51 features respectively, while suboptimal performance can be achieved with 29 (RF) and 38 (SVM) features.

The feature ranking procedure previously performed does not take into account an important factor: the cost of calculation/extraction of the feature. This cost can be both computational (time required to calculate the value of the feature) and monetary (purchase of resources, purchase of licences for third party services). The extraction cost of a feature is to be considered per feature field: if you can obtain the value of a feature then you can obtain all the features of that field. For this reason, the analyses mentioned above were performed also with "feature field" resolution. To this aim, the Wrapper method with Random Forest has been executed with a single feature field at a time. Then, the classification performance has been evaluated increasing the number of feature fields, adding at each step all the features of the best remaining fields, according to the pre-calculated ranking. The results are in Figure 4.7, and show that:

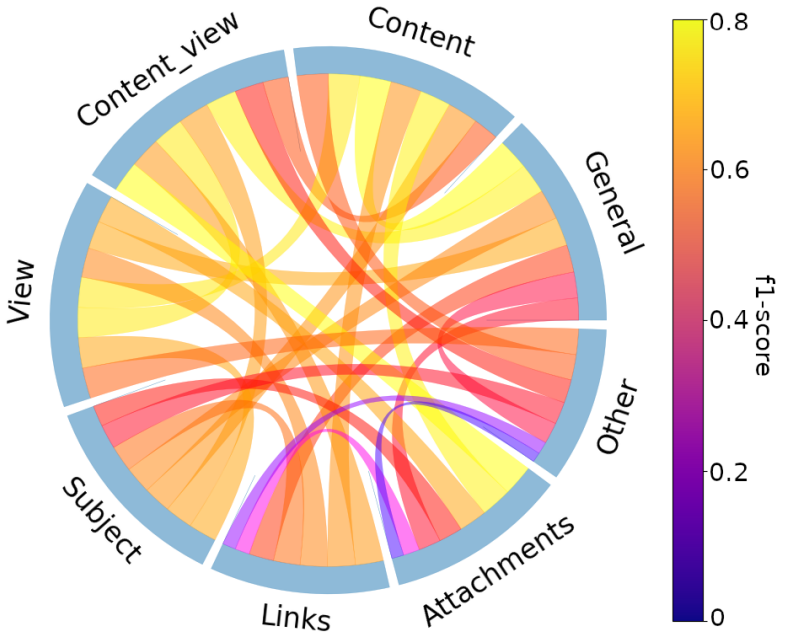


Figure 4.8: f1-score of all possible pairs of feature fields

- as expected the best fields are those concerning the content and the view of the email, thanks to the immediate impact they have on the victims. It also shows that "Content" is redundant when "Content_view" is selected;

- thus not considering "Content", four feature fields are enough to get good performance, avoiding the cost of extracting all the features. However, even just two feature fields could meet minimum requirements and for this reason an exhaustive research on which was the best pair of fields has been done. Figure 4.8 shows the F1-score performance of all possible pairs: the best possible performance with two feature fields can be obtained joining the "Content" features with any of the "View", "General", and "Attachments". This is particularly true for "Attachments" and is an unexpected result, since this field alone has bad performance.

These results are fundamental for deciding which technology and/or service to focus on to develop automated tools for critical spam detection, considering the benefit they bring as a function of their cost.

4.4 Classification performance

Model	Features	Precision	Recall	F1
Random Forest	Full set	0.955	0.909	0.931
	Best 36 Feature	0.952	0.916	0.933
	Best 29 Features	0.933	0.914	0.923
RBF SVM	Full set	0.919	0.871	0.895
	Best 51 Features	0.927	0.880	0.908
	Best 38 Features	0.896	0.885	0.890

Table 4.1: Performance on the test set

The performance evaluation procedure of learning algorithms requires a final test using a set of samples never seen during training and optimization phases. The classification performance of the two models properly tuned were finally tested on the test set (15% of the dataset previously preserved), obtaining the results shown in Table 4.1. The performance is only 1-2 percentage points lower compared to the validation phase. These final results confirm that Random Forest is the best choice for our purposes, with a maximum 95.2% Precision, 91.6% Recall and 93.3% F-Score achieved with 36 features.

Figure 4.9 presents the confusion matrix with the numbers of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). The FP and FN cases were analysed manually, to detect any recurring error patterns and blind spots that could be eliminated. However, by observing and analysing the errors, and also through the experiment described in the next section, we came to the conclusion that there is no precise modelling error that can be eliminated, but the errors are generated by the non fully-deterministic nature of the problem and the resulting labelling. The relevance of a certain email attack is mainly determined by the behaviour of the recipient when the email is received. This causes some emails to be considered irrelevant because that recipient was particularly virtuous (e.g. email immediately reported and no other employee received it), despite the fact that the email had all the characteristics to be successful. Perhaps, for a conservative operator, these “FPs” might not be considered errors, since emails with the same (good) characteristics might reach a less virtuous user the next time. Similarly, some emails despite being of poor quality, some particularly inattentive users were deceived, leading to a positive labelling of these emails. These “FNs” are in fact outliers, gener-

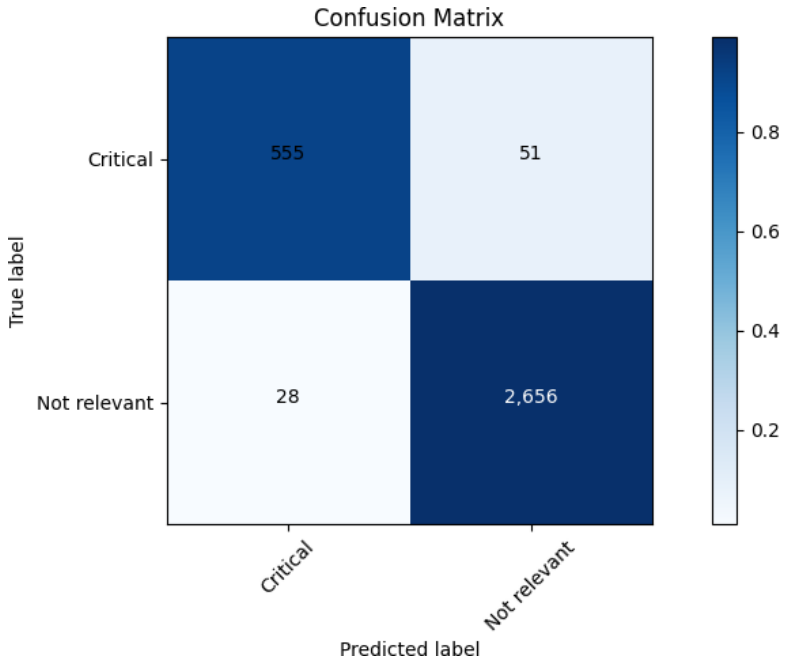


Figure 4.9: Confusion Matrix

ally undetectable with supervised models, for which it might be interesting to complement these models with unsupervised Machine Learning models.

Fortunately, TP and FP are prevalent and such impressive performance values allowed to deploy our automated classifier in the infrastructure of the company, integrating it into the actual email threat management process of the company's SOC. In particular, the classifier analyzes all the reports received by the SOC to prioritize them and highlight the most dangerous ones to the analysts who can then make further investigation to prevent possible incidents and mitigate current ones. The integrated system is now enabling the daily detection of several security incidents that would otherwise go undetected.

4.5 Evading the detector with adversarial samples

The vulnerabilities of machine-learning-based classifiers are well known in the literature [86, 87]. Poisoning attacks are very difficult to perform in our specific scenario because the labeling of samples in the training set is performed manually by a human analyst. Samples that are not manually labeled do not

become part of the training set. On the other hand, evasion attacks are much easier to perform, e.g., through a phishing email that has a perturbed value on specific features considered important by the classifier. These techniques are very effective against image classifiers, notoriously vulnerable because they over-emphasize a small subset of features (pixels). For example, the image of a panda, with some tampered pixels, is still a panda to the human eye, but not for a classifier based on machine learning. *Apruzzese et al.* and *Biggio et al.* show that this is also possible in contexts such as ours and in particular for Random Forest [88] and SVM [89].

It is therefore important to understand if a similar issue also affects the classifier proposed in this work. In particular, it is important to verify if a highly-effective phishing email (i.e. has very good chances of misleading a human and hurting systems) is still effective when its features are perturbed such that it is no longer relevant for the classifier. To this aim, a huge empirical experiment has been set up: an awareness campaign on almost all employees of the company, including top managers and executives. Adversarial samples increasingly distant from the classifier's positive decision region have been generated as synthetic spam emails and sent to the employees over a one week time span. As reported in details in the following, obtained results show that the more the samples enter the negative decision region (and are therefore not detected by the classifier), the less they become effective in succeeding as an attack. The methodology devised for this aim is reported in the following:

1. Clustering of positive samples from the dataset to obtain representative samples of successful attacks. From this procedure about 5 centroids have been obtained, and the most suited one to run the campaign and measure the success rate has been selected: such centroid represents a phishing attack executed with a link. The feature vector from which to generate the adversarial samples has been obtained using this sample

$$f \sim < f_0, ..., f_{m-1} >$$

2. To generate the adversarial samples this feature vector has been manipulated, altering some of the features with a perturbation δ

$$f' = f + \delta \ \sim < f_0, ..., f_i + \delta_i, ..., f_{m-1} >$$

The intensity of the perturbation was appropriately chosen at each manipulation, depending on which feature it was applied to, in order to preserve the integrity of the attack anyway. It is always about 20% of

the value of the feature. The manipulations are 7 (counting also the case of null manipulation) and have been conducted in order to alter the features with more mutual information with the positive class. They are summarized in the Table 4.2. In order to obtain a set of adversarial samples that are less and less relevant to the classifier, each of them is obtained by adding a new manipulation to the previous sample. All possible combinations of manipulations could not be tested in order to avoid sending too many unsolicited emails to the company's people. Let C be the starting centroid obtained by clustering and let S_i be the adversarial samples

$$S_i = C + \sum_{d=0}^{i} \delta^d$$

3. Seven adversarial samples representing the phishing templates used in our experiment have been obtained. Such synthetic emails have been sent to a total of 41,154 people, of all levels of expertise, education and age. Each phishing template reached 5,879 random people. The purpose is to measure the degree of success of each template.

The experiment designed in this way generated the results shown in Figure 4.10, which highlights and confirms that:

- As perturbations increase, the probability of belonging to the positive class (risk score) decreases.
- As the risk score decreases, so does the degree of success of the phishing template. This means that the classifier models the phenomenon correctly.
- The alteration of a feature generates a decrease in the click rate proportional to its importance. The first alterations, made on the most important features, degrade the risk score more than subsequent alterations.

Thanks to this analysis it is possible to identify the best risk score threshold (or δ_{tr} perturbation threshold) in order to prioritize the analysis of email spam reports. For each value of this threshold, which represents the amount of effort that the SOC can provide on this task, the number of possible security incidents detected is maximized, minimizing those that remain unnoticed. This is of fundamental importance due to the impossibility to check all such reports and in order not to waste too much effort on those not dangerous.

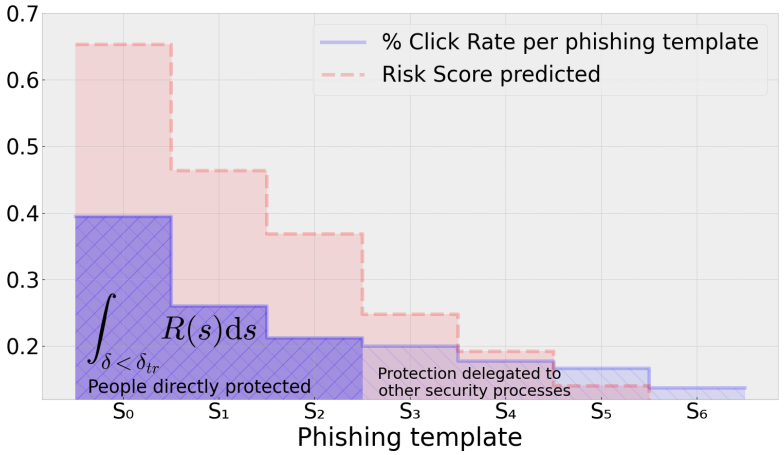


Figure 4.10: Results of the awareness campaign experiment: impact of feature perturbations δ

#	Manipulation	Altered Features
δ^0	No manipulation	None
δ^1	Alteration of the readability of the content by smudging the punctuation	gulpease_index, gulpease_index_clean_text
δ^2	Alteration of the correctness of the content by injecting typing errors	voc_rate(_clean_text), vdb_*_rate(_clean_text)
δ^3	Deletion of hidden text (white text on white background)	hidden_text_*, vdb_*_rate, voc_rate
δ^4	Remotion of deceiving words from the subject	n_scammy, n_phishy, n_*_subject, n_words_subject
δ^5	Dispersion of the deceiving message by adding a long block of text at the bottom of the content and words in the subject line	n_*_subject, n_*_content, screenshot_height
δ^6	Insertion of multiple points where to click by adding clickable images	n_links,vt_l_*, n_images, n_images_links

Table 4.2: Manipulations performed to generate adversarial samples

4.6 Discussion and Limitations

This work shows that with our approach it is possible to automatically distinguish whether a received unsolicited email represents an attack attempt and accurately estimate the probability of its success. In this way, the anti-phishing analysts can be assisted to use their limited resources on the most dangerous phishing attacks. The limitations of our approach and evasion strategies that adversaries might pursue are discussed below.

The infeasibility of analyzing all received emails. Since our model is based on complex features, extracted through long computations and usage of licensed third party services, it is not possible to extend this in-depth security analysis to all received emails, which are millions per day, due to monetary and computational constraints. The feature ranking section (Section 4.3) shows the possibility of feature reduction still saving most of predictive power, enabling computation on much more samples. Unfortunately, the construction of the ground truth through the manual labeling of emails by analysts is not practicable because of privacy issues if the emails are not reported as suspicious. Among the future works, there is the extension of the analysis to all emails using unsupervised approaches. Our supervised approach is therefore built on user reports, thus leading to the next point.

The need of virtuous users. Our approach heavily relies on user reporting, which lead to the engagement of the anti-phishing group on the possible attack received. User involvement in identifying phishing suspects is crucial, as it pre-filters the totality of emails received by the company and selects candidates for a thorough security check. Having a number of users who are aware of these security aspects is therefore an important requirement for achieving a kind of herd immunity that serves to defend themselves and the whole company. With this in mind, it is important to be able to design effective awareness campaigns based on security incidents that affected users in the recent past. The results of feature importance (Section 4.3), from an *Explainable AI* (XAI) perspective, highlight the characteristics of the most impactful email attacks, providing a decisive contribution to tailoring synthetic emails used for awareness campaigns according to the precise vulnerabilities of users, as also demonstrated by the experiment performed, documented above (Section 4.5).

The lack of protection on single-victim attacks. Our approach lacks visibility of single-target successful attacks, because the anti-phishing analysts are only engaged if at least one user who has received the suspected phishing email reports it. This is actually a rare possibility since these types of attacks, in order to increase the probability of success, are almost always launched with

multiple recipients in proper phishing campaigns. However, in the case of single-victim attacks the only possibility is once again to keep users trained to recognise these types of threats, especially for those most targeted by *phishing ad personam* attacks (e.g. top managers, executives and their close collaborators). Client-based tools can also be adopted to support individual users in recognising phishing [90].

Supervised Learning weaknesses. Although the main limitation of supervised approaches has been resolved in this work, namely the need to obtain a labelled dataset, there are other well-known weaknesses to discuss. These include the class imbalance in the dataset; this is one of the problems of the use of supervised approaches [91] in the settings of email security analysis, which exhibit extreme class imbalance (on the order of millions to one). However, our specific approach of processing only reports of phishing suspicions and not all received emails, greatly reduces the class imbalance to the order of 4.5 to 1. In addition, because of overfitting issues, some sub-types of attacks poorly represented (or not represented at all because they were never reported by users) in the positive class may be miss-classified, negatively affecting the Recall performance (about 90%, Section 4.4). Finally, supervised approaches cannot detect 0-day methodologies of phishing attacks. For all these reasons, we plan to also experiment with unsupervised approaches to complement our approach.

Chapter 5

The Human Factor in Phishing

This chapter contains the formalization of the methodology and project choices involved in the design, implementation and testing of a system capable of capturing all human dimensions of the phishing phenomenon. The system has a public interface, a web app, composed of two main parts, each one which serves a specific purpose: a questionnaire (also called *survey*) where users insert important information such as their gender, nationality, age, computer skills, and information about their personality and the test where users are presented with ten e-mails and have to deem each of them as legitimate or phishing. Moreover, there is a final part where users receive a feedback about their performance. The design and the implementation choices will be now described in details, including the lessons learned during the entire project creation lifecycle. The system is publicly accessible at ***https://spamley.comics.unina.it/*** and the collected data is obtainable through the dedicated section on the home page of the website, made available to the scientific community. Note that when subjects expect certain stimuli (i.e., phishing attempts), they tend to adjust their behavior accordingly (i.e., become generally more suspicious of all content), and thus, skew data away from what might be seen in real world settings. In this regard, it is important to reduce the subject's *expectancy effect* [92] in order to collect accurate data on user behavior. Research shows that data accuracy and the subject's expectancy effect in phishing studies is directly related to the realism of the phishing content and the unawareness of the user, that is, not knowing that they will examine phishing content as part of the study [93]. As will be explained shortly, some strategies were adopted in the attempt to reduce such expectancy effect to the minimum possible. The following sections describe the experimental setup,

providing details about the used platform and the rationale of the most important project choices; they also describe the process for the building of the e-mails featured in the experiment.

5.1 The Survey

The first page of the web application is a welcome page featuring a message about the origin of the project, what brought the team to develop it, and a disclaimer stating *"Potentially, all the e-mails you receive could be legitimate, or they could all be phishing: there is total randomness, just like in a real-world scenario."* placed with the purpose of reducing the user *expectancy effect*. Users are then redirected to the survey page, where they are asked questions on their characteristics, ranging from basic information to more specific ones. As a general rule, all the information collected during this phase can be used to define users categories and to analyze the behavior of such categories when faced with phishing attacks. The test is anonymous, which means that it is not possible to do an analysis for a specific person, but the goal of the project is to identify groups of people who behave similarly based on their characteristics. To do so, users have been asked to insert some information about them, selected on the basis of similar work with regard to phishing. The survey has been divided into two sections: one section on general information and one section on psychological traits.
The first section is presented in the following:

- **Basic information**: in this section, users are asked to insert their age, gender, and nationality. According to the selected nationality, the test will propose e-mails in different languages. Specifically, if the nationality is Italian, the test will propose 8 e-mails in Italian and 2 in English, since it is not unrealistic that Italians will receive some communication in English language, while for every other nationality, the test will be composed of 10 e-mails in English language.
  Rationale: according to *Darwish et al.* (2012) [94] *Age* and *Gender* could be two relevant traits when predicting phishing susceptibility (along with *Typology of education*, *Personality traits*, and *Internet usage behavior*).

- **Education**: in this section, users are asked their educational level (e.g. high school diploma, doctorate degree), and their education field (e.g. natural sciences, information technology, law).

Rationale: collecting data on the level and typology of education could reveal very important information on how users' background is associated with vulnerability to phishing attacks. For example, it would be an interesting result if data showed that users with a background education in Science, Technology, Engineering and Mathematics (STEM) subjects do not achieve better results than users with background in non-STEM subjects.

- **Work information and e-mail usage context**: in this section, users are asked about their employment type (e.g. employee, entrepreneur, student), job field (the same options as in "education field"), mailbox usage context (e.g. corporate communications, financial management, shipping), years of job experience, and emails read per day on average. It is worth mentioning that the options in "mailbox usage context" actually match the real life contexts that the e-mails that compose the test refer to.
  Rationale: information about users' work background and usage of the mailbox can be used to determine which user interests are more associated with vulnerability to phishing or to assess whether users that make frequent use of the e-mail service and are familiar with the way these services work inside companies are more resistant to phishing compared to users who do not. For example, employees of technology companies are expected to be active and more accurate with regard to report of potentially dangerous e-mails to the security department. It is worth mentioning that the options in "mailbox usage context" actually match the real life contexts that the e-mails that compose the test refer to. This information can be useful to better understand which users' interests are more correlated with vulnerability to phishing.

- **Skills**: in this section, users are asked to insert a value that indicates a self-assessment of their computer knowledge, a value that indicates the user's self-confidence in spotting malicious e-mails, how many hours they usually spend per day on the Internet, and how many hours they have worked that day before taking the test.
  Rationale: As already discussed, tiredness can play a determining role, also with expert users, hence the inclusion of this last field in the questionnaire. Moreover, determining whether more skilled users are more resistant to phishing has been subject of many studies [62] [94], and it would also be very important to understand which skills are the best at

predicting resistance to phishing and which ones are irrelevant, in order to design tailored solution for prevention.

- **Phishing awareness**: in this section users are asked to specify whether they have ever taken part to anti-phishing courses, and if so, whether they have taken part to anti-phishing courses in the previous six months.
  Rationale: as already discussed, the efficacy of anti-phishing courses is highly debatable, hence the importance of assessing such information is self-evident.

The second section of the survey asks users to perform a self-evaluation on a few aspects of their personality, and psychological traits, with entering a value between 0 and 5, and is organized as follows:

- **Personality characterization**: in this section, users are asked to assign a value to some of their personality traits (e.g. risk perception, risk propensity, concerns on privacy and data).
  Rationale: information of this type can be used to evaluate whether some traits of personality, especially for users with high values, can reveal specific vulnerabilities toward certain typologies of e-mail. For example, it would be interesting to see if users with higher risk perception achieve a higher correct answers rate.

- **The Big 5 Personality Traits**: in this section, users are asked to give an estimate of their five main personality traits, namely agreeableness, conscientiousness, openness, emotional stability and extraversion. In this section there is also a *truthfulness test* in which users are asked to insert the value 3 to demonstrate that they have paid attention while answering the survey. Tests with a truthfulness value different from 3 are marked as invalid and no assumptions about the characteristics of the user are made in the analysis for those tests.
  Rationale: correlation between the big 5 personality traits and phishing susceptibility has been object of many studies, such as [61] and [94].

- **Cognitive Vulnerabilities**: in this section users are asked to indicate which cognitive vulnerabilities they identify in themselves, among scarcity, consistency, authority, social proof, liking and reciprocity, providing a quantitative estimate of their vulnerabilities.
  Rationale: asking users what kind of cognitive vulnerabilities they identify in themselves could provide useful information, for example to evaluate whether there is a difference between the amount of incorrect an-

swers given by users with high values in certain cognitive vulnerabilities with regard to generic e-mails and the amount of incorrect answers given by the same group when answering e-mails that contain the cognitive attack they declare to be vulnerable to.

5.2 The Test

Right after completing the survey, the test starts. Firstly, a prompt asking a name and an e-mail address appears, with the purpose to personalize the e-mails in the test. In this phase, users can use their real name and e-mail address, since it will not be stored in the database. From a technical standpoint, this is achieved by setting to *"NULL"* the relative fields in the database after the result page is *pushed* by the *server* to the *web client*. Right after, the actual test begins and users are presented with 10 e-mails, either legitimate or not and they have to decide, according to their intuition and knowledge, if the e-mail is a phishing or a legitimate one. Optionally, users can also check a box if they would report the e-mail to the security department. Reported mails cannot be deemed as legitimate, for obvious reasons. The simulation includes most of the functionalities of a real e-mail client. Some e-mails also come with a description of the context in which the e-mail is received in; this helps users identify with the situation and respond as they would in a typical usage scenario. After the test, the system stores the following information: which e-mails the user has classified as legitimate, which e-mails the user has classified as phishing, which e-mails the user has reported, the timestamp associated with the conclusion of the test (unfinished tests are discarded), and the time taken for each answer. Having discussed the idea behind the user profiling, the concept behind the construction of the e-mails that appear in the test is now introduced. Previous works on phishing defined what characteristics of a phishing e-mail make it more dangerous, or, to put it more technically, as far as machine learning is concerned, which features have the highest *importance value* when predicting if an e-mail is to be considered potentially dangerous. It is possible to identify two types of e-mail characteristics: automatically extracted text features and manually set labels. Table 5.3 summarizes the various features categorized based on the particular email field which they belong to. In this table it is possible to identify some intrinsic characteristics of e-mails such as number of words, number of phishing-related and scam-related words, number of images, features that carry the information about technical tricks that the attackers can use, such as spoofed domain and phishing links, and fea-

tures that carry information about the psychological tricks employed by the attackers, such as the presence inside the body of persuasion principles. Some examples of sentences Some relating to the persuasion principles are provided in Table 5.1.

Principle	*Phishing e-mail text example*
Consistency	e-mails that include "[RE]" in the subject. "As a follow up to our recent conversation" "The information that you have previously entered is incorrect, please re-enter it",
Social Proof	"as already many users have done this month" "I would like to invite you to join the thousands of other clients who have experienced our vacation excursions".
Reciprocity	"we are doing our best to improve our customer services" "you won this month's lucky gift".
Scarcity	"don't let this opportunity slip away!" "this is a limited edition product" "there are only X left"
Liking	inclusion of sentences that express compliments, praise. "we really appreciate your participation (presence, contribution), therefore we..." "according to your necessities..."
Authority	communications from: government entities; law enforcement officers; your boss; company executive board.

Table 5.1: Application of Principles of Persuasion

Note that the links that appear in the application do not constitute an actual external reference, meaning that users that click on them will not be directed to a new web page, but on the action of *mouse over* a label containing the destination URL will appear. With reference to dictionary-based features: the dictionary containing words associated with phishing consists of a list of 15 words and the dictionary containing words associated with scam consists of a list of 24 words.
The dictionaries with words belonging to the base vocabulary, for English and Italian, contain, respectively: 5,944 words and 7,180 words.The dictionaries with words belonging to the full vocabulary, for English and Italian, contain, respectively: 466,550 words and 986,700 words. The percentage of words contained in the full vocabulary is used as an indicator of syntactical correctness of the e-mail text. The percentage of words contained in the base vocabulary is used as an indicator of text simplicity. The readability index uses the Gulpease index [79] for the Italian language and the Flesch formula [80] for the English language and indicates the level of punctuation adequacy.

FIELD	FEATURE	DESCRIPTION
Sender	sender_wrong	Indicates if the recipient is trying to impersonate someone else.
	sender_type	Indicates whether the recipient's address belongs to a company, person, or other.
Subject	subject_ratio_num	Indicates the ratio between the numbers present in the subject and the total number of characters.
	subject_ratio_chars	Indicates the ratio between the letters in the subject and the total number of characters.
	subject_ratio_special_chars	indicates the ratio of special characters in the subject to the total number of characters.
	subject_forward	Indicates whether the submitted email is a forwarded one.
	subject_response	Indicates whether the submitted email is a reply email.
	subject_noascii_char	Indicates if the submitted email is in ASCII format.
	subject_n_word	Total number of words in the subject.
	subject_n_chars	Total number of chars in the subject.
	subject_n_tags	Number of tags in the subject.
	subject_n_phishy_words	Indicates the total number of phishing words in the email subject line.
	subject_n_scammy_words	Indicates the total number of scamming words in the email subject line.
Body	body_n_words	Total number of words in the body.
	body1_n_words	Total number of words in the first part of the body.
	body2_n_words	Total number of words in the second part of the body.
	body3_n_words	Total number of words in the third part of the body.
	body_n_chars	Total number of chars in the body.
	body1_n_chars	Total number of chars in the first part of the body.
	body2_n_chars	Total number of chars in the second part of the body.
	body3_n_chars	Total number of chars in the third part of the body.
	body_n_phishy_words	It's a dictionary-based feature that indicates the total number of phishing words in the email body.
	body_n_scammy_words	It's a dictionary-based feature that indicates the total number of scamming words in the email body.
	body_voc_rate	It's a dictionary-based feature that indicates the percentage of words in the email body that are present in the vocabulary.
	body_vdb_rate	It's a dictionary-based feature that indicates the percentage of words in the email body that are present in the base vocabulary.
	body_readability	Indicates the level of readability of the text of an email.
	body_lang	Language in which the email is written.
	body_n_pictures	Total number of images in the body.

Cognitive Triggers	body_sense_of_urgency	Indicates whether the email is characterized by a sense of urgency.
	body_reciprocity	Indicates whether the email is characterized by the principle of reciprocity.
	body_consistency	Indicates whether the email is characterized by the principle of consistency.
	body_social_proof	Indicates whether the email is characterized by the principle of social proof.
	body_authority	Indicates whether the email is characterized by the principle of authority.
	body_liking	Indicates whether the email is characterized by the principle of liking.
	body_scarcity	Indicates whether the email is characterized by the principle of scarcity.
Context	context_result	Indicates whether the context is leaning towards legitimate or towards phishing.
Image	body_has_image_in_header	Indicates whether there is an image in the email header.
	body_has_image_in_center	Indicates whether there is an image in the email center part.
	body_has_image_in_footer	Indicates whether there is an image in the email footer.
Signature	body_has_signature	Indicates whether there is a signature in the text of the email.
Greetings	body_has_greetings	Indicates whether there is a greeting in the text of the email.
	body_greetings_custom	Indicates whether the greetings in the body of the email are customized.
Attachments	body_attachment_Maxsize	Maximum size of an attached file.
	body_attachment_Minsize	Minimum size of an attached file.
	body_attachment_doc	Attached file type doc.
	body_attachment_pdf	Attached file type pdf.
	body_attachment_exe	Attached file type exe.
	body_attachment_jpg	Attached file type jpg.
Link	body_links_text	Number of text links in the body. Text links are links that appear as clickable text.
	bottom_links_text	Number of text links in the bottom.
	footer_links_text	Number of text links in the footer.
	body_links_match	Number of match links in the body. Match links are links in which the shown URL (source URL) matches the destination URL.
	bottom_links_match	Number of match links in the bottom.
	footer_links_match	Number of match links in the footer.
	body_links_little_match	Number of little match links in the body. Little match links are links in which the source URL and the destination URL differ by a few characters.
	bottom_links_little_match	Number of little match links in the bottom.
	footer_links_little_match	Number of little match links in the footer.
	body_links_not_match	Number of not match links in the body. Not match links are links where the source and destination URL differ by many characters
	bottom_links_not_match	Number of not match links in the bottom.
	footer_links_not_match	Number of not match links in the footer.

Table 5.3: Email feature set

Each portion or section of an e-mail serves its specific purpose and, it is important to analyze which of these sections of the email affect user perception the most. In the following, further details regarding these sections are provided:

- ***Sender.*** In this category all the general characteristics of the recipient are indicated. Specifically, it indicates whether the sender's domain is incorrect (they are trying to impersonate someone that they are not) and if the email comes from a company, an individual or other.
  Rationale: this feature, although synthetic, has the highest relevance, since spoofing of the domain address is one of the few technical cues of phishing. Assuming the legitimacy of an e-mail just by looking at the sender domain is a matter that requires great experience: a correct e-mail address does not always indicate a legitimate e-mail, since some SMTP servers allow domain spoofing (due to DMARC policies not configured) and the exploitation of this vulnerability is not uncommon, while a wrong sender domain is most of the times a clear evidence of phishing (except for rare cases, e.g. when the communication service of a company is outsourced to a subcontracted company).

- ***Subject.*** This category comprises all the characteristics that define the subject of the email, such as whether it contains non-ASCII characters or it contains phishing-related or scam-related words.
  Rationale: subject lines are the first point of contact with the email recipient and one of the main pieces of information that triggers the user to decide whether or not an email should be opened.

- ***Body***. This category comprises features extracted from the text in the email content.
  Rationale: The body of an e-mail is of fundamental importance, since it holds a great number of indicators involved in the users' decision process when classifying an email as legitimate, phishing, or simply spam/unwanted advertisement. In a typical phishing attack, after the greeting, phishers first describe a problem, then they provide a solution to fix the problem and, finally, they present a link indicator. This is the reason why the body has been divided into three parts: body1 where the problem is described, body2 where the fix is presented, and body3 featuring a link or a button.

- ***Cognitive triggers***. This category carries information on the principles of persuasion present in the content of the e-mail.

Rationale: it has already been discussed how phishers tend to exert certain persuasion principles on users to induce them to comply with their requests.

- ***Context***. Feature that defines the presence of a legitimate or phishing context of the e-mail.
  Rationale: this feature will be described in the next section.

- ***Image***. Features that define whether an image is present in the header, center, footer of the e-mail or not at all.
  Rationale: the images in the email are of great importance; in fact, it is plausible that the main thing that confuses users is the presence of an image itself.

- ***Signature***. Feature that defines if a sender signature is present.
  Rationale: in phishing e-mails, a signature section serves the purpose to confer a formality tone to it. This field is typically present when a phisher wants to use the psychological attack vector of *authority*.

- ***Greetings***. Features that define the presence or absence of the sender's greetings and customized greetings.
  Rationale: The presence of personalized greetings makes the presented email much more similar to a real one; in fact, the phisher generally tends to considerably customize the text of the email in order to make it easier for the user to believe it is legitimate; this strategy is largely used, both in targeted phishing, and in large-scale phishing (for example, the e-mail will result tailored to a large amount of people using a very common name for the greetings personalization).

- ***Attachments***. Features that carry information about the attachment type (e.g. .doc, .exe, .jpg, .pdf) and its size.
  Rationale: Attached files may contain malicious content such as malware, so it is important to analyze them carefully both in size and type. Files containing executable code, such as PDF, should always arouse suspicion.

- ***Links***. These features refer to the links present in the e-mail content.
  Rationale: Phishing emails typically contain a URL that directs users to a fake website, inducing them to disclose important information such as credit card numbers and credentials. Keeping track of the typology of link present in an e-mail can allow to perform interesting analysis.

5.3 The E-mail Data Set

The e-mails featured in our system are composed taking inspiration from real e-mails collected during the activity described in Chapter 3, ranging all kinds of real life domains. A number of strategies that attackers use to create fraudulent emails have been included in the selected list of emails. These include, but are not limited to, spoofing the sender's email address, providing a custom greeting, including sentences that trigger fear and urgency, and mimicking the appearance of legitimate e-mails.
Apart from the numerical features introduced in the previous section, there are a few other e-mail characteristics, which are usually manually set and can be considered as the e-mails meta-data

- ***label***: this field contains a string that indicates which real-life domain is associated with the e-mail (e.g. Sport, Shipping). *Rationale*: this field is useful to determine which real life domain is more associated with phishing susceptibility;

- ***phishing_email***: this value indicates if the e-mail is legitimate or phishing;
  Rationale: the test requires that users receive a set of 10 e-mails, randomly selected among legitimate and phishing;

- ***borderline***: this value indicates whether the e-mail is on the edge between phishing and legitimate. Specifically, if the e-mail is a phishing one, borderline means that, despite showing no clear, technical cues of phishing, taking into consideration context, attachments and the overall appearance of the email, this e-mail should arouse suspicion; if the e-mail is a legitimate one, borderline means that considering context, attachments, links and the overall appearance of the email, this one can arouse suspicion, despite being legitimate. Overall, the possibilities for the labeling are four: phishing not borderline, phishing borderline, legitimate borderline, legitimate not borderline;
  Rationale: sometimes legitimate e-mails put links behind short URLs, which is very suspicious. Sometimes phishing e-mails contain a legitimate link and a malicious attachment; since there's no way to say that an e-mail with an attachment is a phishing attempt just by looking at its appearance, these e-mails are considered borderline, meaning that users must possess a broader knowledge and resort to their lateral thinking skills to correctly classify them. The inclusion of these e-mails in

the pool serves the purpose of testing users ability in recognizing more difficult e-mails.

- ***id***: a value that identifies the e-mail inside the database.
  Rationale: this field can be used as a reference for the ”root e-mail” of a hierarchy of *perturbations* which will be explained next.

- ***parent***: this value, wherever an e-mail was originated from another e-mail applying what goes by the name of a *perturbation*, indicates the ”id” of the e-mail which this one was originated from. A *perturbation* indicates the modification of some details of the e-mail that makes it subtly different in the form but, most of the times, radically different in the meaning associated with it. For example, an e-mail can be legitimate borderline and without a context; by generating a new e-mail from it and giving it a context, the new e-mail might be labeled as phishing borderline, hence changing its nature; alternatively, by changing one letter in the link, the e-mail can go from non-borderline legitimate to non-borderline phishing.
  Rationale: the goal of this mechanism is to find whether a small change determined a change in the user perspective, which would give us precious information about a detail that was determining in making the user fall for the phishing e-mail;

- ***context***: this field indicates a string that explains the situation the e-mail is received in. The context can be a legitimate one if it gives credibility to the e-mail and is coherent with its content or can be a phishing context if it takes credibility away from the e-mail or is discordant with its content.
  Rationale: this field is added to enrich the information contained in the message and to help the user identify with the situation the e-mail is received in. Sometimes the context of a borderline e-mail is the only clue at the user’s disposal to correctly classify it.

Figure 5.1 summarizes the email design. Two separate categories of features can be identified: a set of features related to the fields that compose the e-mail, hence to what the user sees when reading an e-mail, and a set of features that serve as an outline for the e-mail, mostly invisible to the user or serve as a description.

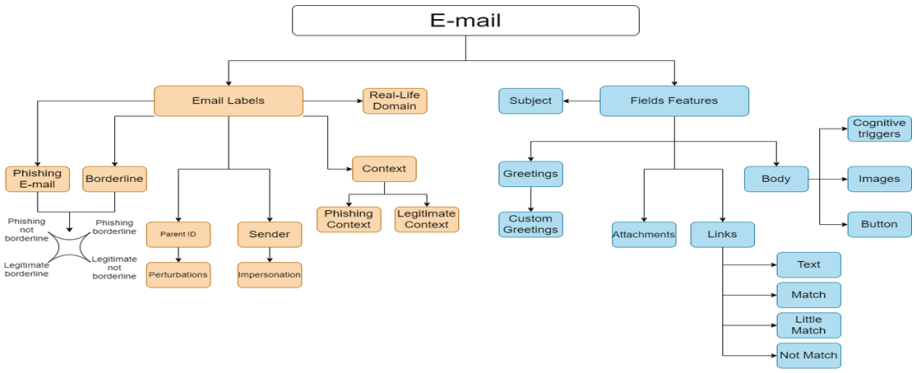


Figure 5.1: Composition of the e-mails

The data set counts, at the moment, a total of 136 unique e-mails, 68 in Italian and their counterparts in English. The number of e-mails is constantly growing and, as the number grows, a higher variety of characteristics of the e-mails is obtained. For each language, there are 42 phishing and 26 legitimate unique e-mails. Each e-mail has either zero, one or more persuasion principles expressed; for phishing e-mails, these represent a cognitive attack, while for legitimate e-mails, they simply represent a strategy for persuading the user. One of the goals of this work is to evaluate whether certain persuasion principles are effective when used in phishing e-mails and to what extent. Therefore, the pool of e-mails has been built in a way that no cognitive attacks are under-represented.

Figure 5.2 illustrates the design example of one of the e-mails that could appear in the test. It is possible to see that the sender address is not immediately visible, but users need to open the screen with the details to see it (by clicking on the button under the sender name). The web client has been designed this way to make it resemble as much as possible the Gmail client, for users to be more familiar with it.

5.4 Feedback Page

The final section of the test features a results page, where users can visualize:

- their overall score, information such as the average response time and a textual description of their score;

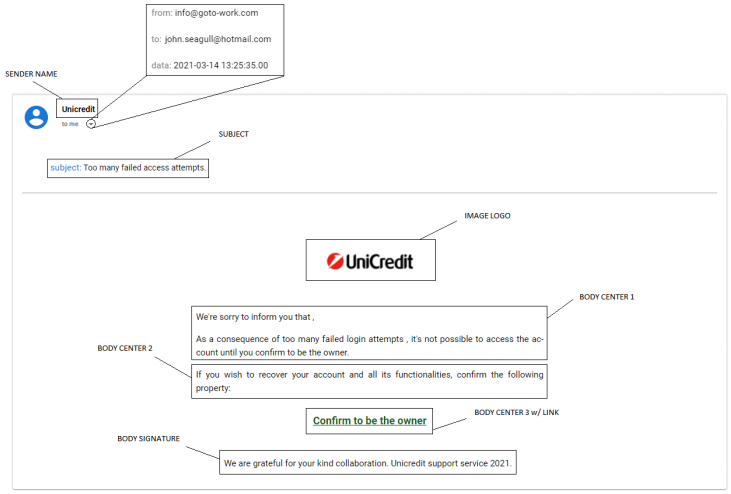


Figure 5.2: Structure of an example e-mail

- a section where users can go in the detail of each e-mail, see which ones they got right, which ones they failed to recognize, which cognitive attacks were present in each e-mail, as well as other more technical features of the e-mail, such as the number of images, the readability index, the simplicity of the vocabulary, the number of words, number of scam and phishing words, number of links etc.

This page is intended to give feedback to the users, who can then inspect their performance, learn from their mistakes and improve the general level of security awareness. Note that, in this phase, in order to reproduce the exact appearance of the e-mail that the user visualized during the test, it is necessary to keep track of the name and the e-mail address that the user inserted right before the test started up until the moment that the result page is sent to the web client. As mentioned, before the test begins, users are invited to feel free to insert their real name and e-mail address for customization purposes, since the application does not store such information; from a technical standpoint, this is achieved by setting to *"NULL"* the relative fields in the database right after the result page is *pushed* by the *server* to the *web client*.

5.5 Analysis of Results

The definition of the analysis that is possible to perform with the data collected through the system is presented in this section. An interpretation of the results achieved by the users that participated in the test is also provided. Results are to be considered preliminary. However, the analysis presents a comprehensive overview of the types of analyses that can be done on the data. The web app was disseminated among companies, universities, schools and also public stands at public events. Participants were chosen in a random way, involving different age groups, educational qualification, computer knowledge, phishing knowledge and so on. As of July 22, the total number of tests taken is 509, although only 412 are marked as valid through of the *truthfulness test* described in *Section* 5.1. For each user, only the first test taken is considered in this analysis, since the ability of users to learn is not the focus of this work. The evaluation setup will be now discussed, then a breakdown of all the conducted analysis is presented.

The Evaluation Setup

As already mentioned, this work has the final goal of finding whether certain characteristics of e-mails can be effective on certain categories of users. In order to achieve that, information on the user's score, information about the users and information on the e-mail have been collected. Four different types of analysis have been created:

- **Analysis 1**: includes analysis of the aggregated score obtained by users without making any assumption on the group they belong to or the characteristics of the e-mail;
- **Analysis 2**: includes analysis of the score obtained by all users, making assumptions on the characteristics of the e-mail;
- **Analysis 3**: includes analysis of the behavior of users divided in groups based on their characteristics, while making no assumptions on the characteristics of the e-mails;
- **Analysis 4**: includes analysis of the score obtained by users divided in groups based on their characteristics and making assumptions on the characteristics of the e-mails.

Analysis 1 - General Behavior

A possible analysis on the results obtained by all users is described in this section with the purpose of analyzing their general behavior. Examples of this typology of analysis are overall percentage of correct answers, overall rate of report to the security department or, as the graph shows in Figure 5.3, the relationship between time taken before answering and the percentage of correct answers. Each test taken has an average time to answer (calculated by averaging the time before answering each of the 10 e-mails that appear in the test). Figure 5.3 shows the Cumulative Distribution Function of such time for the tests that scored less than 6 correct answers on 10 (black plot), scored between 6 and 7 correct answers on 10 (red plot), and scored 8 or more correct answers on 10 (green plot). On 509 total tests, the number of tests that obtained less than 6 correct answers is 142, The number of tests that obtained 6 or 7 correct answers is 242, while 125 tests obtained 8 or more correct answers. These results suggest that 94% of users who scored less than 6 have an average

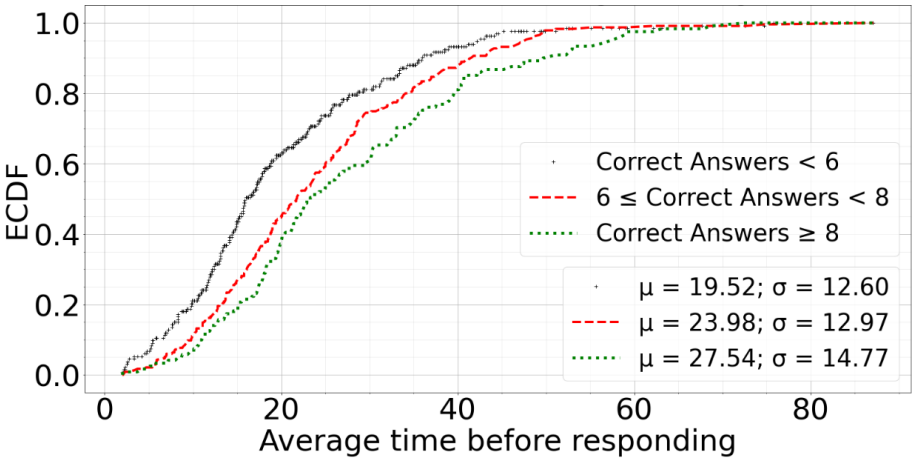


Figure 5.3: Empirical Cumulative Distribution Function of the time taken before answering of users divided by the obtained score. Users having a higher score spend more time to read the e-mails.

response time of less than 40 seconds, 88% of users who scored between 6 and 7 have an average response time of less than 40 seconds, and 81% of users who scored 8 or more have an average response time of less than 40 seconds. This means that the 20% of users who scored more than 8 look at the e-mails for more than 40 seconds, while only 5% of users that scored less than 6 look at the

e-mail for such a time. This could indicate, along with the information on the mean and standard deviation of the three distributions, that users who achieve better scores tend to spend more time reading the e-mail, compared to users who perform under the average (which is 64% correct responses per test). As a result of this analysis, one could derive that an effective prevention technique could include the disabling of links clicking and attachment downloading for a certain amount of seconds after opening the e-mail, to force users to read carefully the e-mail and decrease the probability that they perform incautious actions.

Analysis 2 - Attack Strategies

A possible analysis of the results obtained by users focusing only on e-mails with specific characteristics is described in this section, with the purpose of evaluating the impact of the various techniques employed by phishers and whether certain techniques are more effective than others. Examples of this typology of analysis are: variation of user performance with the real-life contexts of the e-mails, variation of user performance with the number of words of the body or, as shown in Figure 5.4, variation of the performance with the different cognitive attacks.

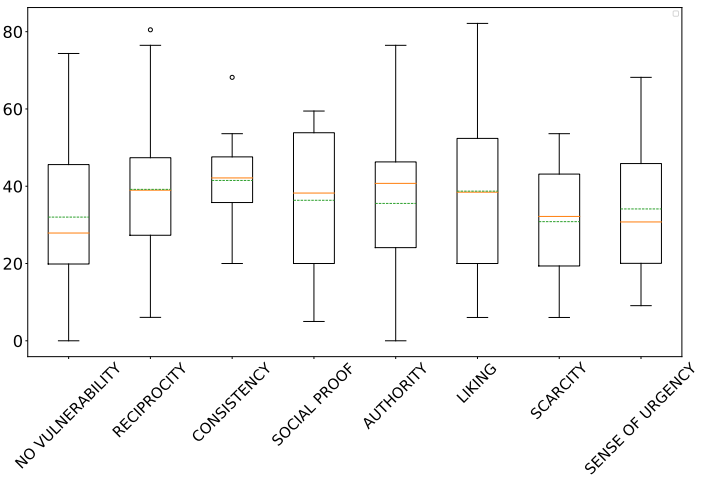


Figure 5.4: Boxplot of incorrectly classified phishing e-mails with respect to different cognitive attacks by all users.

Specifically, users responded to 1387 phishing e-mails without cognitive at-

tacks, 472 phishing e-mails with a *reciprocity* cognitive attack, 516 phishing e-mails with a *consistency* cognitive attack, 409 phishing e-mails with a *social proof* cognitive attack, 658 phishing e-mails with an *authority* cognitive attack, 439 phishing e-mails with a *liking* cognitive attack, 356 phishing e-mails with a *scarcity* cognitive attack, 1010 phishing e-mails with a *sense of urgency* cognitive attack. This result could prove valuable as a guide for the design and implementation of awareness campaigns specialized on specific cognitive vulnerabilities.

Analysis 3 - General Behavior of User Groups

A possible analysis of the results obtained by users divided in groups based on their characteristics is described in this section. This analysis has the purpose of evaluating the possibility that certain categories of users have lower classification accuracy when analyzing content which they do not know the nature of. Analysis 3 and 4 rely on users' accurate completion of the initial survey, therefore, only samples validated by the *truthfulness test* are considered in the subsequent analysis. Examples of this typology of analysis are: variation of score of users with different age, score of users with respect to their gender or, as shown in Figure 5.5, the distribution of reports of users based on whether they have an occupation in STEM subjects or non-STEM subjects. Specifi-

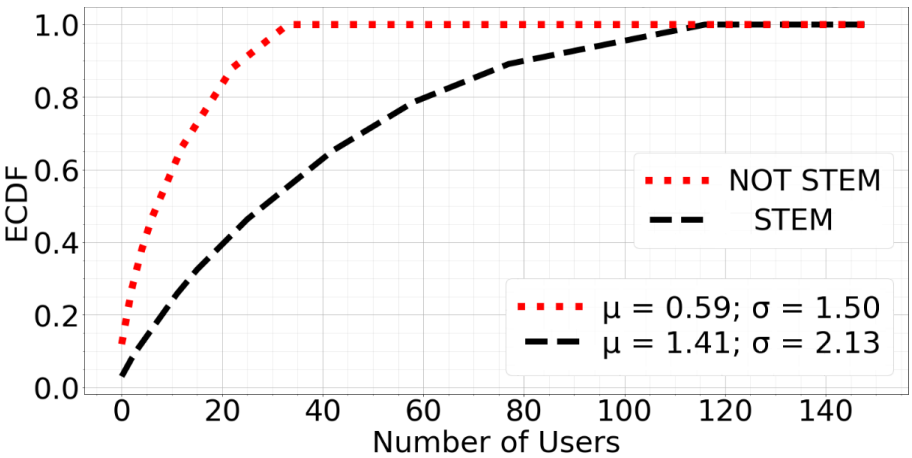


Figure 5.5: Empirical Cumulative Distribution Function of e-mails reported to the security department by users divided by their job type. People working in the STEM field tend to report more e-mails.

cally, there are 256 users with a job type in STEM subjects which reported to the security department 283 e-mails and 156 users with a job type in non-STEM subjects that reported 55 e-mails.
It is possible to see that 115 STEM users reported an e-mail while only 33 non-STEM users reported an e-mail to the security department. This result could highlight the necessity to perform specialized training for users who do non-STEM jobs who might be completely agnostic towards the important practice of e-mails reporting, crucial for the cyber security of a company. One way of doing this might include training employees on what a Security Operation Center (SOC) is, what happens when a user reports an e-mail to the SOC, the benefits of a cooperative framework (described in Chapter 3, and so on.

Analysis 4 - Specific Behavior of User Groups

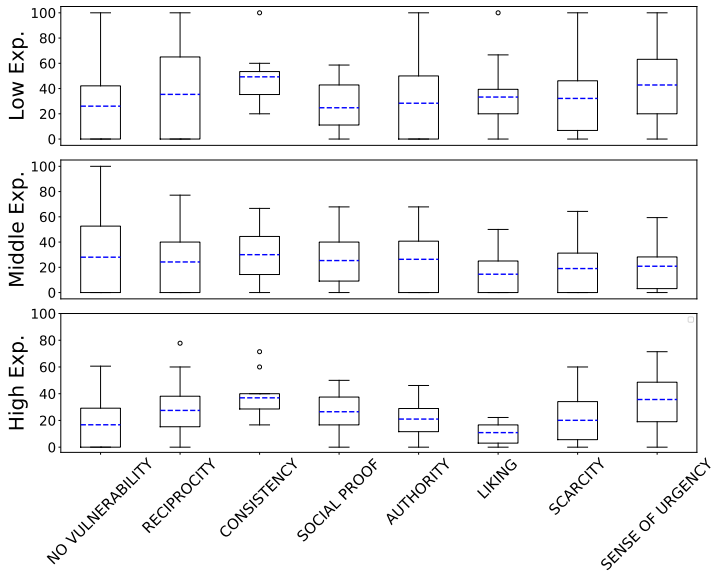


Figure 5.6: Boxplot of incorrectly classified phishing e-mails with respect to different cognitive attacks by users divided by their job experience. People having less job experience are more vulnerable to authority attacks

Finally, the analysis that incorporates the most ambitious goal of this work is the one that make assumptions on both the characteristics of the e-mails and the characteristics of users, with the purpose to evaluate whether some categories

of users are more vulnerable than others when confronted with a specific type of phishing e-mail. This analysis is of prime relevance, since it allows to devise ad-hoc awareness solutions for users; however, since these analysis require to perform multiple filters on the data (on both the users and the e-mails) also require the largest number of completed test; therefore, especially for this case, results are to be considered as preliminary. Clearly, the more data the system collects, the stronger the assumptions that can be made about user behavior. Examples of this typology of analysis are all the possible combination of analysis of typology 2 and 3. Figure 5.6 is an example of such analysis that shows the results obtained by users categorized by their job experience when facing phishing e-mails with the authority cognitive attack. For example, looking at the distributions associated to the *Authority* attack it is possible to see that the effectiveness of such attack decreases as the job experience of users increases. In the upcoming section will be provided a discussion on the possible developments of the platform and on possible future work to increase the value of the data collected.

5.6 Discussion and Future Work

This work presented a platform for evaluating users' ability to distinguish between legitimate and phishing e-mails and a preliminary analysis of this data. The potential of this platform to profile users and evaluate the way they respond to different phishing strategies commonly employed by phishers makes it a powerful tool that, over time, could reveal unforeseen behavior of users. All the knowledge present in the literature about phishing studies has been gathered and a framework that shares most of their goals and findings has been designed and deployed. The need for greater security awareness is clear and this platform has the goal of facilitating awareness campaigns by providing useful information on the behavioral traits of many different categories of users. Since many users who participated in the test stated that they have never received phishing awareness campaigns, the *Spamley* Platform has fulfilled the dual role of a tool for the collection of data useful for scientific research and a means to introduce the phenomenon of phishing to many people, all over the world.

Future development of the application could involve the increasing of the variety of the e-mails with respect to their characteristics, possibly by implementing an automatized mechanism that generates new e-mails obtained through a perturbation of the statically-defined ones. Furthermore, after obtaining a

sufficient number of diverse e-mails, a useful feature might include a mechanism that takes into consideration the real-life context that the user uses the e-mail service for and performs an intelligent matching of the e-mails that appear in the test. Further works on this project could imply the developing of a system that, with the users' permission, saves the e-mail addresses provided by those who have completed the test in the simulated environment and sends fake phishing e-mails to their real mailbox; in this way, it will be possible to do the same analysis carried in the simulated environment, but in a circumstance where the expectancy effect is totally absent. To increase the accessibility of the application we plan on creating multiple versions of the e-mails used in the test in different languages, such as Spanish, French, German, Arabic, Chinese and much more. In this regard, we heartily invite all the members of the scientific community to contact us if you wish to collaborate with us on this ambitious project. To capture the *ocular behavior* of users when reading an e-mail, a study with an on-screen gaze tracking device could be conducted, in a way that, by analyzing the differences between expert users and non-expert users in terms of the areas that are analyzed first and the amount of time each area is analyzed, it could be possible to derive a behavior profile to be taken as an example for training less experienced users. A more sophisticated approach for the user profiling with regard to psychological traits could make use of practices described in the psychology literature, such as *psychometric tests*, which can provide a quantitative estimate of the psychology of the user with a very high accuracy (an IQ test is an example of psychometric test, as well as one of the oldest and most reliable). Finally, after collecting a sufficient amount of data, it will be possible to devise "smart" e-mail clients that provide tailored solution for each user in order to decrease the number of clicks on phishing e-mails. For example, if the data stated that users who are more resistant to phishing read e-mails more thoroughly, it could be considered an option to disable clicking on links or downloading attachments for a certain amount of seconds after opening the e-mail, or if users with low job experience fall more often into e-mails with authority (which is also a claim of other studies on phishing), then it could provide a notification when the user receives an e-mail that contains a phrase expressing a sense of authority, stating "be careful, this e-mail could mislead you".

It is safe to believe that the value that this work provides will increase with time, since the more data it collects, the more it will be possible to make accurate statements on users' behaviors. In general, defining the way that the human mind operates when reading e-mails could be the key to understanding

the best way to protect users from falling into phishing. To this end, this work focused on gathering as much information as possible on the background of users, in order for it to support future research on the human factor involved in phishing and for the devising of new techniques for phishing prevention.

Conclusions

Email attacks are such a commonly used vehicle for the perpetration of subsequent attacks, representing a major threat that affects all industries and causes significant harm. Anti-spam filters do not solve the problem of cyber attacks by spam emails, which still succeed in spreading malware, stealing confidential data, and generating large illicit profits. For this reason, companies typically rely on teams of security analysts to perform manual inspection on such emails. However, spam emails that evade spam filters, especially in the case of large companies, are too many for such analysis to be effective. In this thesis we aimed at providing a contribution to this important problem.

In early 2018, we have built a collaborative framework that collects spam emails and supports the labeling of the actually dangerous ones as critical, through the continuous monitoring of analysts. Using this labeled dataset we have shown that machine learning algorithms can well classify emails as critical, highlighting the threats. The obtained massive experimental results show that Random Forest achieves the best performance, with 95.2% Precision, 91.6% Recall, and 93.3% F-measure. The impact of different feature sets on such performance has been analyzed (Chapter 4.3). Results show that the best performance can be obtained with a selection of the best 36 features out of 79. Since the extraction cost of a feature is shared among the ones of the same type, they have been grouped into sets referred as *feature fields*. Performance has also been evaluated while varying the number of feature fields: by using 4 out of the 8 feature fields, which results in a significant cost reduction, performance degrades by (only) 5%. The feature ranking work also provides an important explanation on how critical emails are built and can be detected. This knowledge led to the design of a week-long awareness campaign, which involved all 40,000+ employees of the partner company, including top managers and executives (Chapter 4.5). This large social experiment confirms that our system correctly models the phishing phenomenon and, together with well-trained people, represents a global defence ecosystem robust

to the majority of email attacks. Thanks to the results obtained and lessons learned, we re-engineered the email threat management process of the partner company around this collaborative approach. It now relies on experienced and aware users who report suspicious emails, an automatic data collection and analysis system, and security analysts who investigate in depth according to the system's suggestions. The presented framework is easily replicable and deployable also in any other organization with a Security Operation Center. We believe that our contributions can lead to a greater awareness of the risks faced by companies and, above all, to the automation of the detection of threats in spam emails.

In the final part of this work we presented a system to help fighting the issue of e-mail phishing totally focusing on the human factor. Our system has the dual aim of disseminating awareness among users and collecting and sharing data regarding user behavior when reading e-mails. A deep analysis of the state of the art we presented in this thesis shows that the focus of studies on phishing has more and more shifted from technical to human-oriented aspects. We have presented the design and implementation of our system that is made of three main components: a web application to test user awareness about phishing, a survey to identify the most interesting characteristics of users, and a large and varied set of test emails incorporating the several possible cognitive vulnerabilities of phishing e-mails. The potentiality of this project to profile users and evaluate the way they respond to different phishing strategies commonly employed by phishers makes it a powerful tool that can reveal unforeseen behavior of users. Data that our system is collecting can be valuable in the devising advanced solutions to the phishing problem. We envisage the creation of "smart" e-mail clients that provide tailored solution for each user, e.g. disabling clicking on links or downloading attachments for a certain amount of seconds, or providing a warning when inexperienced users receive e-mails that contain an authority attacks. The value that this work provides to the scientific community will increase over time. In general, defining the way that the human mind operates when reading e-mails will be the key to finding the best solution to phishing. Our work focused on gathering as much information as possible to support current and future research on the human factor in phishing. The system and the data are publicly accessible at ***https://spamley.comics.unina.it/***.

Ethical Considerations

The experiment described in Section 4.5 was approved and conducted together with the company's security and internal communication departments. Best practices concerning ethics and phishing experiments were used [95]. Although this experiment contravenes widely accepted informed consent requirements and involves deception, we conducted it with extreme care for privacy and confidentiality. No results were associated with the identity of individuals during the analysis, and the training course is run automatically for users who have not recognized phishing. The experiment poses no real risk, since the links and attachments sent do not carry any real threat. Potential participants have the opportunity to opt out of this kind of experiments, and are debriefed after the end of each session.

The research project described in Chapter 5 was carried out under strict privacy policies with regard to data acquired from users. In fact, the test is conducted on a voluntary basis, and it is not possible, in any way, to trace the identity of the user from the stored data. In this regard, anonymity is guaranteed by means described in Section 5.2. The dataset of e-mails used for the test has been composed using historical e-mails and, being a simulated environment, users are not exposed to any real risk, since links in the application do not behave as external references and no real e-mail was sent during the experiment.

Bibliography

[1] Symantec. 2019 internet security threat report. https://www.symantec.com/security-center/threat-report, 2019.

[2] David Piscitello Greg Aaron, Lyman Chapin and LLC. Dr. Colin Strutt. Interisle Consulting Group. Phishing landscape 2021. https://www.interisle.net/PhishingLandscape2021.pdf, 2021.

[3] Special Agent Vicki D. Anderson FBI Cleveland. Fbi warns of rise in schemes targeting businesses and online fraud of financial officers and individuals. https://www.fbi.gov/contact-us/field-offices/cleveland/news/press-releases/fbi-warns-of-rise-in-schemes-targeting-businesses-and-online/-fraud-of-financial-officers-and-individuals, March 2016.

[4] Federal Bureau of Investigation Internet Crime Compliant Center. 2019 internet crime report, Feb 2020.

[5] EUROPOL EC3. Spear phishing, a law enforcement and cross-industry perspective. https://www.europol.europa.eu/sites/default/files/documents/report_on_phishing_-_a_law_enforcement_perspective.pdf, November 2019.

[6] Jianjun Chen, Vern Paxson, and Jian Jiang. Composition kills: A case study of email sender authentication. In *29th USENIX Security Symposium (USENIX Security 20)*, Boston, MA, August 2020. USENIX Association.

[7] Andrzej Duda Sourena Maroofi, Maciej Korczynski. From defensive registration to subdomain protection: Evaluation of email anti-spoofing schemes for high-profile domains, May 2020.

[8] J. D. Ndibwile, E. T. Luhanga, D. Fall, D. Miyamoto, G. Blanc, and Y. Kadobayashi. An empirical approach to phishing countermeasures through smart glasses and validation agents. *IEEE Access*, 7:130758–130771, 2019.

[9] Robert W. Shirey. Internet Security Glossary, Version 2. RFC 4949, August 2007.

[10] Amber van der Heijden and Luca Allodi. Cognitive triaging of phishing attacks. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1309–1326, Santa Clara, CA, August 2019. USENIX Association.

[11] Luca Allodi, Tzouliano Chotza, Ekaterina Panina, and Nicola Zannone. On the need for new antphishing measures against spear phishing attacks. *IEEE Security & Privacy*, PP, 09 2019.

[12] Cisco. Cybersecurity threat trends report. `https://umbrella.cisco.com/info/2021-cyber-security-threat-trends-phishing\ \-crypto-top-the-list`, 2021.

[13] Robert Cialdini. Influence: The psychology of persuasion. 01 1993.

[14] Luigi Gallo, Alessandro Maiello, Alessio Botta, and Giorgio Ventre. 2 years in the anti-phishing group of a large company. *Computers & Security*, 105:102259, 2021.

[15] Luigi Gallo, Alessio Botta, and Giorgio Ventre. Identifying threats in a large company's inbox. In *Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks*, Big-DAMA '19, pages 1–7, New York, NY, USA, 2019. Association for Computing Machinery.

[16] Sally Hambridge and Albert Lunde. DON'T SPEW A Set of Guidelines for Mass Unsolicited Mailings and Postings (spam*). RFC 2635, June 1999.

[17] Oliver Hohlfeld, Thomas Graf, and Florin Ciucu. Longtime behavior of harvesting spam bots. In *Proceedings of the 2012 Internet Measurement Conference*, IMC '12, pages 453–460, New York, NY, USA, 2012. Association for Computing Machinery.

[18] Matthew B Prince, Benjamin M Dahl, Lee Holloway, Arthur M Keller, and Eric Langheinrich. Understanding how spammers steal your e-mail address: An analysis of the first six months of data from project honey pot. In *CEAS*, 2005.

[19] Chia Yuan Cho, Juan Caballero, Chris Grier, Vern Paxson, and Dawn Song. Insights from the inside: A view of botnet management from infiltration. *LEET*, 10:1–1, 2010.

[20] Brett Stone-Gross, Thorsten Holz, Gianluca Stringhini, and Giovanni Vigna. The underground economy of spam: A botmaster's perspective of coordinating {Large-Scale} spam campaigns. In *4th USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET 11)*, 2011.

[21] Christian Kreibich, Chris Kanich, Kirill Levchenko, Brandon Enright, Geoffrey M Voelker, Vern Paxson, and Stefan Savage. Spamcraft: An inside look at spam campaign orchestration. In *LEET*, 2009.

[22] Christian Kreibich, Chris Kanich, Kirill Levchenko, Brandon Enright, Geoffrey M Voelker, Vern Paxson, and Stefan Savage. On the spam campaign trail. *LEET*, 8(2008):1–9, 2008.

[23] Gianluca Stringhini, Oliver Hohlfeld, Christopher Kruegel, and Giovanni Vigna. The harvester, the botmaster, and the spammer: On the relations between the different actors in the spam landscape. In *Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security*, ASIA CCS '14, pages 353–364, New York, NY, USA, 2014. Association for Computing Machinery.

[24] Kirill Levchenko, Andreas Pitsillidis, Neha Chachra, Brandon Enright, Mark Felegyhazi, Chris Grier, Tristan Halvorson, Chris Kanich, Christian Kreibich, He Liu, Damon McCoy, Nicholas Weaver, Vern Paxson, Geoffrey M. Voelker, and Stefan Savage. Click trajectories: End-to-end analysis of the spam value chain. In *2011 IEEE Symposium on Security and Privacy*, pages 431–446, 2011.

[25] Chris Kanich, Christian Kreibich, Kirill Levchenko, Brandon Enright, Geoffrey M Voelker, Vern Paxson, and Stefan Savage. Spamalytics: An empirical analysis of spam marketing conversion. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 3–14, 2008.

[26] Chris Kanich, Nicholas Weaver, Damon McCoy, Tristan Halvorson, Christian Kreibich, Kirill Levchenko, Vern Paxson, Geoffrey M Voelker, and Stefan Savage. Show me the money: Characterizing spam-advertised revenue. In *20th USENIX Security Symposium (USENIX Security 11)*, 2011.

[27] Adam Mossoff. Spam - oy, what a nuisance. *Berkeley Technology Law Journal*, 19:625, 2004.

[28] Federal Bureau of Investigation Internet Crime Compliant Center. Business e-mail compromise the 12 billion dollar scam, July 2018.

[29] A.A. Orunsolu, A.S. Sodiya, and A.T. Akinwale. A predictive model for phishing detection. *Journal of King Saud University - Computer and Information Sciences*, 2019.

[30] Anti-Phishing Working Group INC. (APWG). Phishing activity trends report. `https://docs.apwg.org/reports/apwg_trends_report_h1_2017.pdf`, 2017.

[31] Florian Quinkert, Martin Degeling, and Thorsten Holz. Spotlight on phishing: A longitudinal study on phishing awareness trainings. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 341–360. Springer, 2021.

[32] Pieter Agten, Wouter Joosen, Frank Piessens, and Nick Nikiforakis. Seven months' worth of mistakes: A longitudinal study of typosquatting abuse. In *Proceedings of the 22nd Network and Distributed System Security Symposium (NDSS 2015)*. Internet Society, 2015.

[33] Panagiotis Kintis, Najmeh Miramirkhani, Charles Lever, Yizheng Chen, Rosa Romero-Gómez, Nikolaos Pitropakis, Nick Nikiforakis, and Manos Antonakakis. Hiding in plain sight: A longitudinal study of combosquatting abuse. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 569–586, 2017.

[34] Florian Quinkert, Tobias Lauinger, William Robertson, Engin Kirda, and Thorsten Holz. It's not what it looks like: Measuring attacks and defensive registrations of homograph domains. In *2019 IEEE Conference on Communications and Network Security (CNS)*, pages 259–267. IEEE, 2019.

[35] Nick Nikiforakis, Marco Balduzzi, Lieven Desmet, Frank Piessens, and Wouter Joosen. Uncovering the use of homophones in domain squatting.

[36] Grant Ho, Asaf Cidon, Lior Gavish, Marco Schweighauser, Vern Paxson, Stefan Savage, Geoffrey M Voelker, and David Wagner. Detecting and characterizing lateral phishing at scale. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 1273–1290, 2019.

[37] A. Bhadane and S. B. Mane. Detecting lateral spear phishing attacks in organisations. *IET Information Security*, 13(2):133–140, 2019.

[38] David B. Buller and Judee K. Burgoon. Interpersonal Deception Theory. *Communication Theory*, 6(3):203–242, 03 2006.

[39] Judee K. Burgoon, David B. Buller, Laura K. Guerrero, Walid A. Afifi, and Clyde M. Feldman. Interpersonal deception: Xii. information management dimensions underlying deceptive and truthful messages. *Communication Monographs*, 63(1):50–69, March 1996.

[40] Anirudh Ramachandran and Nick Feamster. Understanding the network-level behavior of spammers. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 291–302, 2006.

[41] Olivier van der Toorn, Roland van Rijswijk-Deij, Bart Geesink, and Anna Sperotto. Melting the snow: Using active dns measurements to detect snowshoe spam domains. In *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*, pages 1–9. IEEE, 2018.

[42] Basheer Al-Duwairi, Ismail Khater, and Omar Al-Jarrah. Detecting image spam using image texture features. *International Journal for Information Security Research*, 3, 12 2013.

[43] Enrico Blanzieri and Anton Bryl. A survey of learning-based techniques of email spam filtering. *Artificial Intelligence Review*, 29(1):63–92, Mar 2008.

[44] Yuli Dai, Shunsuke Tada, Tao Ban, Junji Nakazato, Jumpei Shimamura, and Seiichi Ozawa. Detecting malicious spam mails: An online machine learning approach. In *Neural Information Processing*, pages 365–372, Cham, 2014. Springer International Publishing.

[45] Jason Chan, Irena Koprinska, and Josiah Poon. Co-training on textual documents with a single natural feature set. pages 47–54, 01 2004.

[46] Eirinaios Michelakis, Ion Androutsopoulos, Georgios Paliouras, George Sakkis, and Panagiotis Stamatopoulos. Filtron: A learning-based anti-spam filter. In *Proceedings of the 1st conference on email and anti-spam. Mountain*, 2004.

[47] Emmanuel Gbenga Dada, Joseph Stephen Bassi, Haruna Chiroma, Shafi'i Muhammad Abdulhamid, Adebayo Olusola Adetunmbi, and Opeyemi Emmanuel Ajibuwa. Machine learning for email spam filtering: review, approaches and open research problems. *Heliyon*, 5(6):e01802, 2019.

[48] Wilfried Gansterer and David PÃ¶lz. E-mail classification for phishing defense. pages 449–460, 04 2009.

[49] Basavaraju Mallikarjunappa and Dr R. Prabhakar. A novel method of spam mail detection using text based clustering approach. *International Journal of Computer Applications*, 5, 08 2010.

[50] Isredza Rahmi A Hamid and Jemal H Abawajy. An approach for profiling phishing activities. *Computers & Security*, 45:27–41, 2014.

[51] Areej Alhogail and Afrah Alsabih. Applying machine learning and natural language processing to detect phishing email. *Computers & Security*, 110:102414, 2021.

[52] Panagiotis Bountakas, Konstantinos Koutroumpouchos, and Christos Xenakis. A comparison of natural language processing and machine learning methods for phishing email detection. In *The 16th International Conference on Availability, Reliability and Security*, pages 1–12, 2021.

[53] J. Postel. Simple Mail Transfer Protocol. RFC 821, August 1982.

[54] Bo Holst-Christensen and Erik FrÃ¸kjÃ¦r. Security issues in smtp-based email systems. In *2021 14th CMI International Conference - Critical ICT Infrastructures and Platforms (CMI)*, pages 1–6, 2021.

[55] Scott Kitterman. Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1. RFC 7208, April 2014.

[56] Murray Kucherawy, Dave Crocker, and Tony Hansen. DomainKeys Identified Mail (DKIM) Signatures. RFC 6376, September 2011.

[57] Murray Kucherawy and Elizabeth Zwicky. Domain-based Message Authentication, Reporting, and Conformance (DMARC). RFC 7489, March 2015.

[58] Vladimir V Riabov. Smtp (simple mail transfer protocol). *River College*, 2005.

[59] P. Campbell, B. Calvert, Cisco Learning Institute, and S. Boswell. *Security+ Guide to Network Security Fundamentals*. Thomson/Course Technology, 2003.

[60] Frank Stajano and Paul Wilson. Understanding scam victims: seven principles for systems security. *Communications of the ACM*, 54(3):70–75, 2011.

[61] James L Parrish Jr, Janet L Bailey, and James F Courtney. A personality based model for determining susceptibility to phishing attacks. *Little Rock: University of Arkansas*, pages 285–296, 2009.

[62] Rachna Dhamija, J Doug Tygar, and Marti Hearst. Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 581–590, 2006.

[63] Daniele Lain, Kari Kostiainen, and Srdjan Capkun. Phishing in organizations: Findings from a large-scale and long-term study. *arXiv preprint arXiv:2112.07498*, 2021.

[64] Ponnurangam Kumaraguru, Justin Cranshaw, Alessandro Acquisti, Lorrie Cranor, Jason Hong, Mary Blair, and Theodore Pham. School of phish: A real-world evaluation of anti-phishing training. 01 2009.

[65] Pavlo Burda, Tzouliano Chotza, Luca Allodi, and Nicola Zannone. Testing the effectiveness of tailored phishing techniques in industry and academia: a field experiment. In *Proceedings of the 15th International Conference on Availability, Reliability and Security*, pages 1–10, 2020.

[66] Simone Pirocca, Luca Allodi, and Nicola Zannone. A toolkit for security awareness training against targeted phishing. In *International Conference on Information Systems Security*, pages 137–159. Springer, 2020.

[67] Ahmed Aleroud and Lina Zhou. Phishing environments, techniques, and countermeasures: A survey. *Computers & Security*, 68:160 – 196, 2017.

[68] Grant Ho, Aashish Sharma, Mobin Javed, Vern Paxson, and David Wagner. Detecting credential spearphishing in enterprise settings. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 469–485, Vancouver, BC, August 2017. USENIX Association.

[69] Gianluca Stringhini, Manuel Egele, Apostolis Zarras, Thorsten Holz, Christopher Kruegel, and Giovanni Vigna. B@bel: Leveraging email delivery for spam mitigation. In *Presented as part of the 21st USENIX Security Symposium (USENIX Security 12)*, pages 16–32, Bellevue, WA, 2012. USENIX.

[70] Asaf Cidon, Lior Gavish, Itay Bleier, Nadia Korshun, Marco Schweighauser, and Alexey Tsitkin. High precision detection of business email compromise. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1291–1307, Santa Clara, CA, August 2019. USENIX Association.

[71] Ponnurangam Kumaraguru, Steve Sheng, Alessandro Acquisti, Lorrie Cranor, and Jason Hong. Lessons from a real world evaluation of anti-phishing training. pages 1 – 12, 11 2008.

[72] Luca Allodi Pavlo Burda and Nicola Zannone. Don't forget the human: a crowdsourced approach to automate response and containment against spear phishing attacks.

[73] M.L. McHugh. Interrater reliability: the kappa statistic. *Biochemia Medica*, 22:276–282, 2012.

[74] Ted Humphreys. State-of-the-art information security management systems with iso/iec 27001: 2005. *ISO Management Systems*, 6(1):15–18, 2006.

[75] Dr. Sarwat Nizamani, Nasrullah Memon, Mathies Glasdam, and Dong Duong Nguyen. Detection of fraudulent emails by employing advanced feature abundance. *Egyptian Informatics Journal*, 15, 08 2014.

[76] Natalie Sappleton and Fernando LourenÃ§o. Email subject lines and response rates to invitations to participate in a web survey and a face-to-face interview: the sound of silence. *International Journal of Social Research Methodology*, 19(5):611–622, 2016.

[77] Jaclyn Wainer, Laura Dabbish, and Robert Kraut. Should i open this email? inbox-level cues, curiosity and attention to email. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 3439–3448, New York, NY, USA, 2011. Association for Computing Machinery.

[78] RSA. *Understanding Indicators of Compromise (IOC) Part I*, 2012.

[79] Pietro Lucisano and Maria Emanuela Piemontese. Gulpease. una formula per la predizione della difficolta dei testi in lingua italiana. *In Scuola e Citta (3)*, pages 57–68, 1988.

[80] R Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233, 1948.

[81] Matthias Lee. Pytesseract https://pypi.org/project/pytesseract, 2014.

[82] S. M. Lee, D. S. Kim, J. H. Kim, and J. S. Park. Spam detection using feature selection and parameters optimization. In *2010 International Conference on Complex, Intelligent and Software Intensive Systems*, pages 883–888, Feb 2010.

[83] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3):1–21, 03 2015.

[84] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014. 40th-year commemorative issue.

[85] Quanzhong Liu, Chihau Chen, Yang ZHANG, and Zhengguo Hu. Feature selection for support vector machines with rbf kernel. *Artif. Intell. Rev.*, 36:99–115, 08 2011.

[86] Ling Huang, Anthony D. Joseph, Blaine Nelson, Benjamin I.P. Rubinstein, and J. D. Tygar. Adversarial machine learning. In *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*, AISec '11, pages 43–58, New York, NY, USA, 2011. Association for Computing Machinery.

[87] Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317 – 331, 2018.

[88] G. Apruzzese and M. Colajanni. Evading botnet detectors based on flows and random forest with adversarial samples. In *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*, pages 1–8, 2018.

[89] Battista Biggio, Igino Corona, Blaine Nelson, Benjamin I. P. Rubinstein, Davide Maiorca, Giorgio Fumera, Giorgio Giacinto, and Fabio Roli. *Security Evaluation of Support Vector Machines in Adversarial Environments*, pages 105–153. Springer International Publishing, Cham, 2014.

[90] Melanie Volkamer, Karen Renaud, Benjamin Reinheimer, and Alexandra Kunz. User experiences of torpedo: Tooltip-powered phishing email detection. *Computers & Security*, 71:100 – 113, 2017.

[91] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[92] Vivek Anandpara, Andrew Dingman, Markus Jakobsson, Debin Liu, and Heather Roinestad. Phishing iq tests measure fear, not ability. In *International conference on financial cryptography and data security*, pages 362–366. Springer, 2007.

[93] Peter Finn and Markus Jakobsson. Designing ethical phishing experiments. *IEEE Technology and Society Magazine*, 26(1):46–58, 2007.

[94] Ali Darwish, Ahmed El Zarka, and Fadi Aloul. Towards understanding phishing victims' profile. In *2012 International Conference on Computer Systems and Industrial Informatics*, pages 1–5. IEEE, 2012.

[95] Finn Peter R Resnik David B. Ethics and phishing experiments. *Science and Engineering Ethics*, 24(4):1241–1252, 2018.