International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03121
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Data Drive: A OCR Model to Prevent Phishing

Authors:
Shreyas Rajak, Dr. Muneeswaran V, Dr. Subhash Chandra Patel

Abstract

Phishing is a deceptive tactic used by cyber-criminals to extract sensitive information such as passwords or credit card information etc. from the targeted persons by impersonating the real website, through fake emails/ whatsapp or text messages, sometimes also via images or other files, among others. Most phishing detection methodologies will scan website URLs or text content of emails in order to identify threats. However, in instances where the phishing content is hidden within images, these methods prove inefficient because visual data cannot be parsed appropriately. That's why this paper proposes an OCR-based, data-driven scheme for the automatic extraction and analysis of text from images for this technique to detect phishing whenever there is malicious content within the visual representations. This approach enhances the overall detection accuracy because it focuses on image-based text to offer improved protection against phishing threats.

This proposed model uses OCR for automatically reading text into data from an image and then processes the data using machine learning algorithms to detect other common indicators of phishing, such as suspiciously formed URLs or brand names without authorization. This model's target is the gap that conventional phishing detection systems lack and improves their accuracy by extending protection to cases where text is embedded in images.

Testing on a range of phishing images has shown that the model was able to classify a phishing attempt using good accuracy. Thus, such an OCR-based system, as concluded in this study, further establishes how it can strengthen the security perspective with detection at a better level. It indicates that such a system can be effectively used in other digital platforms, and eventually prevent advanced phishing attempts in the minds of users.

Keywords: Phishing Detection, Optical Character Recognition (OCR), Data-Driven Approach, Cybersecurity, Image-based Phishing, Machine Learning, Text Extraction, Visual Content Analysis, Cyber Threat Detection, Anti-Phishing Techniques.

Chapter 1: Introduction

1.1 Background

In the digital world, the internet imposes tremendous influence on every constituent of life, starting from communication and commerce towards education and entertainment. But in most of such advancement comes a heavy price as serious security issues. Cybersecurity is one of the most imperative issues in the digital world because it aims at safeguarding systems, networks, and programs from digital assaults. Phishing is the most common and destructive methods of cyber attacks carried out by online criminals. Therefore, phishing attacks target individuals, pretending to be legitimate communication from well-known organizations, which results in the theft of sensitive information such as passwords, credit card numbers, andother personal details.

Indeed, most phishing attacks come in many forms. Some can be presented in emails that pose as crucial entities and request user credentials. Sometimes, it appears like a link where, upon clicking, the victim is redirected to a fake website resembling the original one. These attackers always engage their social engineering skills to manipulate the victim into thinking that they are dealing with a valid one. Apart from targeting an individual account, phishing also targets an organization. Data breaches of this kind bring immense losses in the money books, could attract huge lawsuits, and damage the reputation hugely.

Initially, phishing was very simple as attackers just sent emails containing lousy grammar, weird requests, and full of dubious links. As technology grew, phishing attacks became more sophisticated. E-mails, websites, as well as other forms of phishing communication have evolved so much to the extent that even advanced, professional users can't determine whether such an e-mail came from a particular institution. Attackers can do designs for phishing that are very similar to the legitimate emails that appear to originate from trusted institutions such as banks, government agencies, or companies.

To counter such threats, researchers in cybersecurity have invented different kinds of detection mechanisms. Generally, detection had traditionally been approached through techniques designed to scan URL addresses of websites, addresses of email senders, or text content as potential threats. Such an approach is generally effective in cases of phishing attacks using the possibility of text or direct links. For example, if an email contains a suspected URL, the system may mark it as dangerous and send an alert warning users not to click the link. But lately, the attackers have become more creative by embedding the phishing content inside images, which makes the traditional text-based scanning methods much more difficult to detect.

It is much tougher when the phishing content is laid inside an image or other graphical components. For example, the phishing attackers might send messages with logos or other

graphical elements that imitate valid organizations, with fake e-mails or messages. Because the malicious intent is well hidden within the image, typical phishing detection systems, which are mostly based on textual analysis, are likely to fail to detect such attacks. The images might also include text containing images to develop it in a manner that it appears to be just a normal part of the content, like developing a phishing login page within an image, and it becomes totally impossible for regular systems to identify the phishing.

In current times with image-based phishing gaining huge momentum, more complex detection algorithms become essential so that they can also treat as well as process the visual content. It is at this point where the existence of OCR technology plays a role. OCR gives computers the ability to get text from images and process them, thereby enabling computers to analyze the very content on images for a potential phishing threat. It will be easy to take text in images, convert it into machine-readable format, hence make possible broader analysis about the content, with OCR technology.

1.2 Why OCR in Phishing Detection

To this point, the fact that image-based phishing attacks have increased has shown us that text-based analysis alone is no longer enough to produce effective phishing detection. Most attackers embed malicious content in images in the form of screenshots of false login pages, forged documents, or fake logos-all so as to cheat users. The images can be fantastic in appearance and are designed in a way that it closely resembles a legitimate communication. Since the traditional phishing detection systems focus mainly on textual content, they are unable to fully parse and analyze the visual data, resulting in a significant gap in the detection process.

One of the most critical functions of OCR technology in filling this gap is automatic text recognition from images or scanned documents, and changing the text to a machine-readable format. In such cases, the difference it makes for detection within images about phishing is extremely important because it helps the system "read" text contained within an image and analyze it for possible phishing signs.

Applying OCR in phishing detection implies extracting the suspicious text from images attached to e-mails or websites, or any other form of communication. For instance, an attacker might attach a false login page inside an image. OCR would then extract this text for comparison against legitimate login pages. The phishing detection system would then trigger on the inconsistencies such as a false URL, unauthorized usage of brand names, or modified text indicating a phishing attempt.

Another very important advantage of applying OCR in phishing detection is the possibility of integrating OCR with machine learning models, thus greatly improving the accuracy of

detections throughout the process. Once the text is extracted from the image, then it can be further processed by an algorithm designed to recognize certain phishing patterns. Such models can pick up on suspicious keywords, incorrect formatting, or even more indicators of phishing, thus greatly enhancing the system's ability to identify and classify phishing attempts.

For example, it could be fed typical phishing indicators; these are URLs that look dodgy in structure, such as misspelled brand names; phrases or words that have, until recent times, been common amongst phishing emails, such as "urgent" or "your account suspended"; or the unauthorized usage of branded logos. The detection of phishing becomes much broader and broader, along with analysis of both the text itself and the elements of the communication- even the visual aspects - made possible by combining such machine learning techniques with OCR.

OCR is the most critical need in the fight against phishing and is required for all cases involving phishing content hidden inside images. While extracting text from images and further processing with some form of machine learning-based algorithm gets this to an overall higher accuracy and effectiveness in detecting phishing systems, this reduces the gap between traditional textual forms of detection methods and these emerging image-based phishing tactics.

1.3 Problem Definition

Phishing attacks have evolved long ago with attackers finding multitudes of ways to avoid traditional detection. Among the more challenging approaches in phishing detection is the use of images to conceal malicious content. Indeed, the old-fashioned text-based phishing attacks that involve scanning URLs cannot be detected by those traditional phishing detection systems that look for text-based content; most phishing attempts still go undetected.

One more complexity added to this is that these image-based phishing attacks are designed to very closely resemble legitimate communications. It could possibly include mal-login form, unauthorized brand logos, or some misleading text embedding within an image. These users will have no chance of distinguishing between a legit and a malicious content. Conventional phishing detection systems do not recognize the image-based phishing attempts. Such attacks have been designed so they do not bypass the text-based analysis techniques.

To overcome this, an advanced system must be developed to detect an image-based phishing attack in an automated mode. Specifically, this research envisions the development of a data- driven, OCR-based detection system designed primarily for text extraction and analysis from images. The proposed system will close this gap by using the OCR technology to convert the text within the images into a format readable by the machine, hence yielding better protection for image-based phishing threats.

1.4 Objectives

The primary objective of this research study is to develop a phishing detection system able to identify phishing efforts disguised in the form of images by means of Optical Character Recognition technology using machine learning algorithms. The objectives of the researchare:

1. **Develop an OCR-based phishing detection system**: In this purpose, the objective is to have a system that will utilize OCR to extract text from images and analyze any retrieved text for phishing indicators.
2. **Machine learning models should be integrated**. The overall second aim is to incorporate such algorithms into the system in order to enhance the phishing detection accuracy. Such models will be trained on two basic elements, namely, phishing-related patterns and indicators, including suspicious URLs, unauthorized brand names, common phishing phrases, among others.
3. **To test the performance of the system**: The third objective is to test the effectiveness of the proposed system on different phishing images. Evaluation will be conducted with a point of view for the accuracy, precision, and recall values of the system for phishing.

It also aims at strengthening the detection capabilities for phishing. Closing the gap between traditional text-based approaches and image-based phishing tactics will improve the overall process of phishing detection.

1.5 Scope and Limitations

This research focuses on developing an OCR-based phishing detection system that can detect phishing attempts that are embedded within images. Here, the scope of the study encompasses designing and implementation as well as evaluation of the system, where specific emphasis is drawn toward the application of OCR technology and machine learning algorithms. The scopes of the research areas include the following:

1. **Image-based phishing detection**: This research predominantly focuses on the detection of images found in phishing attacks. That involves phishing email attacks, fake websites and phishing attacks coming from social media with embedded malicious content in the images. From those images, text is extracted and analyzed to infer if any type of phishing has occurred in the system.
2. **System implementation**: The proposed system is essentially based on the application of OCR technology. OCR allows the system to extract text from images, an activity

that otherwise traditional phishing detection methods would not identify visually. Therefore, the research focuses on applying OCR technology in phishing detection and integrating it with the machine learning models.

3. **Machine learning integration**: The system integrates algorithms that can identify phishing with a better degree of precision. Such algorithms are also given a pattern that characterizes phishing-like suspicious URLs, common wording in phishing, and unauthorized use of a brand names.
4. **Evaluation and testing**: The paper involves an evaluation of the system's performance by testing it on a given set of phishing images. The performance of the system will be measured in terms of its precision, recall, and accuracy in performing during the phishing attack.

Despite the effective approach there are some limitations found in this research:

- Limited to text-based phishing attacks: The given system is specifically tailored to authenticate and validate the phishing attacks, which are based on the text attached to images. This system might fail at detecting purely graphical content-based phishing attempts because it has no texts.
- Dependency on OCR accuracy: The phishing detection system largely depends upon the effectiveness of the OCR technology adopted for it. If the OCR system fails to extract text from an image, then it may compromise the phishing detection process.
- Small dataset: The experiments are conducted based on a specific set of phishing images. Although the dataset is huge but still in general, the system might need further fine-tuning for satisfactory performance on other datasets or in real-world applications, where the differences between phishing images could be significant regarding content as well as quality.

1.5 Scope and Limitations

Conclusion The paper has scoped the study with respect to an enhancement of phishing detection capabilities through OCR technology and ML. Even though the system has multiple limitations, it still represents a great step forward in detecting phishing attempts incorporated into images and is a significant contribution in cyber security research overall.

1.6 Structure of Thesis

The structure of this thesis is below: Chapter 1 (Introduction) provides background and motivation for the study, defines a problem statement, sets forth the aims and objectives of the book, lists the key research contributions, and explains the organisation of the thesis. Literature Review Chapter 2 encompasses an overview of phishing detection techniques; it also encompasses the evolution of OCR technology, data-driven approaches for phishing detection, and identifies gaps in the existing literature. Methodology Chapter 3 presents the proposed system that focuses on the OCR process by providing a detailed explanation of Tesseract OCR followed by the various phases for the image preprocessing, feature extraction, and classification models used for the study. The chapters describe the implementation details of the model, evaluate its performance, and experiments carried out with the accuracy metric. Chapters 5 (Discussion) analyzes the results, compares them with related work, and discusses limitations. Finally, in Chapter 6 (Conclusion and Future Work), the main results will be summarized, and the directions for future work will be elaborate with an emphasis on the practical outcomes of the work.

Chapter 2: Literature Review

2.1 Overview of Phishing Detection and Previous work

Phishing is one of the most common cybercrime methods that, over time, has shaped into one of the most preponderant online security threats. It refers to the fake act of posing as a trusted party aimed at obtaining private information such as usernames, passwords, credit card numbers, and personal identification numbers. This is usually sent via email but can also be sent through instant messaging, social media, or even a fake website. Given that most daily activities will be on the use of online applications, phishing attacks have become a common threat both to people and organizations. There are four broad categories of techniques for detecting phishing, namely: content-based detection, heuristic-based detection, blacklist- based detection, and machine learning-based detection.

Twenty related work has been studied thoroughly and we have observed that machine learning methods, including Decision Trees, Random Forests, and Support Vector Machines, have shown their effectiveness in phishing URL detection by attaining high accuracy with minimal false positives and false negatives [1][2[3]. Further, the approaches are such as K-means clustering that find better performance than the other methods since these maximized accuracy and efficiency in identification of phishing sites at the time of maintaining the low false positive and false negative ratios [4][5]. The integration of feature extraction with machine learning can offer promising prospects toward the detection of phishing websites that actually overcome the shortcomings of traditional approaches as well as enhancing the possibility to identify fresh phishing threats in time [6][7][8]. In another study, it has been suggested to detect phishing messages by analyzing the content of URLs as well as webpage content and deleting them automatically if classified under phishing [9]. Another approach analyzes the contents using images and detects phishing attempts through image recognition. The brands' logos will be recognized, and the source domain cross-checked with authentic domains of well-known brands [10][11]. The other approach allows for the discovery of unknown phishing websites from large repository websites by analyzing the IP addresses of confirmed phishing sites and identifying adjacent suspected domains, which are then compiled into a list and further analyzed on phishing URLs [12]. Through machine learning, phishing detection systems have also emerged based on an email's content and sender information combined with website features which successfully identify and block phishing attacks [13]. Other techniques use distinctive information and risk evaluations that consist of automatic detection and manual detection processes that focus on phishing sites suspected to pose malicious activity [14]. Real-time detection of a phishing attack includes increased precision with the aid of techniques such as forms submission verifications, logo extraction, and root name

comparisons against whitelists, at real-time performance [15][16]. Machine learning-based approaches such as Adaboost training a classifier using URL and website features to enhance accuracy through classification confidence [17]. In addition to the rule-based comparison of URLs that emphasizes the path content, keyword ordering has been used to effectively detect phishing pages [18]. Content-based, heuristic-based, and fuzzy rule-based methods help to distinguish phishing websites from legitimate sites to protect against social engineering attacks [19]. Another method is based on the comparison of visual and functional similarities between phishing and legitimate websites. Users may be enabled to contrast between authentic and fraud web sites [20]. Lastly, a new model uses image visualization of website code and extracts features from malicious URLs, further enhancing the effectiveness and accuracy of phishing detection. The model assesses the website addresses by calculating the similarity and security risk coefficients, analyzing the characteristics of the web page to well identify the phishing sites [21].

2.1.1 Content-Based Detection:

The content-based detection methods analyze the actual content of an email, message, or a webpage to determine attacks on phishing. This can include inspection of keywords in text for suspicious incidences of words or phrases, a search for specific features in a URL, or even checks on the format and layout of a message. Some of the phishing emails use elements of legitimacy and maliciousness together to give false impressions regarding official logos and sender addresses but embed fraudulent links. But as attackers have become sophisticated and mimicked legitimate entities, content-based detection systems are facing issues in detecting slight variations in legitimate and phishing communications. As attackers change the content to escape detection most of the time, this system has also lost its effectiveness.

2.1.2 Heuristic-Based Detection:

The second category is heuristic-based detection, which exploits predefined rules or patterns to determine phishing attempts based on some characteristics. Most of these are made by learned phishing website or email behavior. They will include such indicators as having an IP address in the URL instead of a domain, URL encoding/decoding techniques, and even differences between the domain name of an email and the content of the email. One big advantage of heuristic-based methods is that they can detect as-yet-unknown phishing websites or emails that have not yet appeared in the blacklists. They are likely to result in false positives in case the heuristics are too wide. Moreover, phishing attackers usually adopt quickly, hence launching phishing campaigns that look nothing like typical patterns; these methods will be only short-lived effective.

2.1.3 Blacklist-Based Detection:

Blacklist-based detection is one of the most widely deployed systems. It works by keeping a database of known phishing sites, IP addresses or email addresses that have been pegged as malicious. When the system detects an incoming e-mail or website, it checks this against the blacklist. When a match is found, the system may either block an email or just disallow a user from visiting a phishing site. Blacklisting is quite effective when phishing sites are detected early on and quickly added to the blacklist. However, phishing sites may live only for a few hours before they are gone offline or replaced by others. Thus, blacklist-based systems cannot track new or changing phishing sites that have not yet been reported.

2.1.4 Machine Learning-Based Detection

Machine learning (ML) and artificial intelligence (AI) technologies have become increasingly important in phishing detection as of the recent past with fast-paced advancement in those technologies. Machine learning models are trained in datasets of phishing and legitimate emails or websites to learn the usual patterns typical of phishing. These models can process very broad types of features, for example, the URLs' structure, language usage in emails, characteristics from the sender, and details about the registration of the domain. Once trained, these models are able to classify new emails or websites as phishing or legitimate accordingly. The efficiency towards the sophisticated and evolving phishing tactics is their adaptability of ML-based systems for the change in time using the learning from new attempts of phishing. Deep learning models, like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have greatly improved phishing detection capabilities of a model in identifying the subtle patterns of phishing within the emails or websites. However, a machine learning-based detection is highly dependent upon the quality and diversity of the training data set. Secondly, the models may have poor detection of novel attacks that differ significantly from the previously known phishing pattern detection. The various emerging forms of attacks that phishing has had to adapt to include spear phishing-targeted and whaling targeting high-profile individuals. One of the existing openings in phishing attacks is to mobile users and social media/instant messaging apps, less covered than traditional detection systems. Attackers now insert phishing links into multimedia formats, such as image and PDF and videos, and there it gets difficult to detect. As this phishing evolves further, there exists an urge for new methods to detect these threats effectively in non-text-based environments also. OCR technology can be used since it is text extraction from images and other files with documents besides other forms of multimedia presentations for analysis. This way, hidden phishing content in non-text formats can come to light.

2.2 OCR Technology

Optical Character Recognition, shortly denoted as OCR, refers to technology intended for extracting and interpreting text from images, scanned documents, or other visual formats. OCR converts this visual content into machine-readable text that can be analyzed and manipulated by a variety of systems. From its original use in the 1920s as support for the visually challenged, OCR has been through major leaps forward over the last century. For purposes of broad distinction, the three stages include rule-based OCR, statistical OCR, and deep learning-based OCR.

1. Rule-Based OCR: The first OCR system made extensive use of rule-based systems where it relied entirely on the pixel recognition to match specific pixel formations with predefined templates of characters. Such systems performed pretty well with printed text when the font types are consistent and of good quality scans. But fails with variations in font, size, handwriting or distorted texts. The early OCR systems were incredibly rigid in its ability to generalize beyond specific patterns programmed into them. For instance, these systems performed very poorly if the scanned text was a handwriting text, or it was a low-quality document where characters did not fit into predefined templates. Therefore, rule-based OCR systems had limited uses and were more or less confined to environments with controlled formats, such as processing printed invoices or checks.

2. Statistical OCR: The introduction of statistical methods in OCR marked a giant leap. Instead of relying purely on pattern recognition, statistical OCR systems based their probable predictions on models that forecast the most likely character following the image. That allowed OCR to accept a greater range of text formats, fonts, and even certain levels of handwriting input. Techniques like Hidden Markov Models (HMM) were used for processing character sequences, hence improving the ability of the system to recognize words and phrasesrather than single letters.

Statistical OCR shared this advantage in growing computational power. Now it was possible to store large data sets of scanned text and provide these systems with more accurate and robust representations. Further, due to its feature of handling mixed case letters and noisy or distorted text, statistical OCR was more versatile compared to earlier ones that strictly depended on rules. However, it also proved to be much sensitive to highly complex documents along with mixed content in images containing images or tables overlay with text since the statistical models could easily lose track.

3. Deep Learning-Based OCR: Precision and flexibility have been introduced to the latest generation of OCR systems with improved machine learning and deep learning technology. Deep learning models, particularly CNNs, have greatly improved OCR accuracy. This was found to be quite impressive since CNNs have proved to be highly efficient in image processing with pattern recognition and thus perfect for extracting text from noisy or complex visual data. This is unlike earlier OCR models, which have relied on hand-defined features entirely for their earlier success; CNN-based models can automatically learn relevant features from the data, making them more likely to adapt to new forms of text, different fonts, and even distorted or handwritten texts. Deep learning OCR algorithms can now be used to overcome challenges that have long proven elusive for older OCR techniques: skew, variable sizes, text being written on the most complex backgrounds possible, and low-quality images. For example, one of the most popular OCR tools, Google's Tesseract, achieves high accuracy on many document types and formats using deep learning algorithms. The deep-learning- trained OCR models can also process multilingual text and hence can be applied globally. In phishing detection, OCR comes up as the central part since it helps extract text messages embedded into images, PDF, or other such multimedia formats that the traditional phishing detection system might not be able to examine. For example, attackers embed phishing URLs or malicious messages in images, so that text-based detection mechanisms may not find the same. One may employ OCR followed by the feeding of extracted text to machine learning algorithms for presence in a phishing attempt. The newest applications of OCR involve integration with other AI-based systems, thereby making the system much better. For example, modern methods of NLP may be able to recognize the context in which the text had been extracted, for use in phishing detection through recognizing suspicious language patterns or phrases that often feature in other phishing attempts.

Applications and Challenges of OCR in Phishing Detection:

It can be especially useful for scanning such e-mails or websites with OCR, specially ones formatted in non-standard ways like PDFs, images, or any graphical-based content. Sometimes the phishing e-mails are going to carry the pictures of the text sometimes coming with the format of login pages or other deceiving messages. This text can be captured by the OCR and passed on for further analysis carried out by the phishing algorithms-which may either be machine learning classifiers or heuristic-based methods.

Despite all these developments, OCR technology still encounters several barriers with this regard of application. It largely hampers the precision when applied on low-quality images or skewed texts, or for any intricately configured structure of a document. Furthermore, even a hacker can extend his ploy to such mechanisms within phishing attacks to blur the content of the phishing content and avoid accuracy in extractions performed by OCR systems. It further

faces a challenge, as well because OCR systems require substantial resource computation, especially if these are applied in real-time detection of phishing where response has to be made within a short time.

With the most modern approach, nowadays, sophisticated multi-lingual or non-Latin scripts phishing attacks are fast becoming the order of the day in modern phishing campaigns. Some OCR systems are capable of recognizing non-Latin scripts but are generally less accurate compared to Latin-based text. The need for phishing detection systems to operate worldwide imposes a challenge on their capability to identify an important phishing attempts by considering multiple languages and scripts.

2.3 Data-Driven Methods in Phishing Detection

Phishing detection methods shall be data-driven: They need large amounts of data, sophisticated algorithms, and a sharp eye in spotting phishing attempts. Common sense would tell that the inherent meaning of these methods is that even though phishing attacks come in all shapes, flavors, and sizes, any phishing attack would share some common features or characteristics that can be learned from data. It lies at the core of modern phishing detection where machine learning, including deep learning, is said to empower systems to detect known phishing tactics and adapt to new phishing tactics as they come.

1. Machine Learning-Based Phishing Detection:

The widespread use of phishing detection models relies on the fact that machine learning models can generalize large datasets. Usually, the most common types of supervised learning employed are decision trees, support vector machines, and random forest classifiers. The models are trained on a labeled dataset where each instance comes with a 'phishing' or 'legitimate' label attached to it. Once this model is developed, it can recognize new, unseen emails or websites as either phish or legitimate by learning patterns from data. For instance, Fette et al. (2007) had developed a relatively early phishing detection system that utilized a range of machine learning algorithms; the system was trained on a dataset comprised of email headers, body content, and URLs. The technique demonstrated the ability of machine learning algorithms to classify phishing emails with nearly perfect accuracy compared to traditional heuristic-based methods. Since then, a range of variants has emerged incorporating even much more complicated features and strong models of machine learning.

2. Features Used in Machine Learning Models

Feature selection is one of the biggest problems in machine learning-based phishing detection. Features are almost the variables or characteristics that describe an email or website used as input to the machine learning model. Some features commonly available in phishing detectioninclude

- URL Characteristics Some of the URL characteristics include whether it is long or short, if it contains suspicious keywords such as "login," "secure," or "verify," and whether it contains unusual characters such as "@", "-", or an IP address. These types of URLs make it difficult to identify phishing sites. It is able to allow machine learning to determine patterns.
- Email Content Features, that is, what is written in the body of the email. For instance, phrases or patterns, which can usually be found within phishing messages, like "urgent action necessary," asking people to provide personal information or threats about closing an account. Most NLP techniques are utilized in finding keywords for phishingmessages.
- Sender Information like all most phishing e-mails are originated from spoofed accounts or one does not know. The machine learning model can identify the domain name of the sender's email address and match up with a list of known legitimate domains.
- They are designed to mimic the legitimate sites but have different structural and design features. For example, the security certificates may be missing in phishing sites, or they may use HTTP instead of HTTPS or low-quality images with broken links. The machine learning models may use such features to separate the legitimate sites from phishing sites.

3. Deep Learning Techniques in Phishing Detection

The last years have been dominated by deep learning models in phishing detection. This ranges from CNN to RNN models used in complex features of phishing emails and websites. Deep learning models automatically learn features from raw data, eliminating the need for manually pre-selected features.

For instance, CNNs can be adapted easily to scanning a web layout or in an e-mail because it can perceive it as just another image. Since CNNs are hierarchical, patterns they will recognize

range from the character level up to very high-level structures such as full URL or even full web page layouts. The primary application of RNNs when it comes to the sequential behavior that may be perceived when scanning text in emails or within URLs, is actually in using Long Short-Term Memory networks. Given that the LSTMs can actually capture long-term dependencies in text, they would especially be useful in determining a possibly valid decision about whether phishing likely occurred based on the flow of language in an email.

A very popular study is that of Bahnsen et al. (2017) by applying LSTM networks for detection of phishing from URLs. In their model, the sequence of characters of the URLs was used, and with such, they could distinguish a phishing site with a very high precision even when the URLs are obfuscated in order to avoid detection.

4. OCR and Machine Learning in Phishing Detection :

So, all of this makes PDFs, images, or videos a vector to embed phishing content, and this is all relevant due to OCR for data-driven phishing detection. Images can be leveraged with phishing e-mails or websites containing text-terms as in a fake login prompt or a malicious URL, for instance, as an attempt to evade detection by text-based detection systems. OCR can extract the text that might be embedded in the image and feed it into machine learning models.

OCR can even include machine learning in order to take phishing detection to the highest level. For example, the organization's logo can be in the phishing mail along with the CAPTCHA image containing the text written on it. The traditional phishing detection system would fail at this stage to detect the phishing attempt but an OCR can extract the text and can forward the text to the machine learning classifier for further processing.

For example, Mohamed et al. (2020) used OCR integrated with machine learning in the identification of phish in images. They applied OCR to extract text from the embedded images of phishing mails. Then using the classification model using machine learning, they classified the extracted text as either phishing or genuine. Their system correctly classified a phishing email composed of an embedded image, and truly, the combination of OCR with machine learning really proved its effect in the detection of phishing emails.

5. Data Augmentation and Training Models:

Data-driven phishing detection systems rely heavily on large, labeled datasets for their training. However, obtaining phishing datasets is very difficult because, by the time a study can be performed, phishing site addresses have often been changed. Further, frequent updates

are required to capture all the possible phishing addresses. To overcome such difficulties, researchers make use of data augmentation techniques that artificially increase their training datasets. Data augmentation involves generating new training examples based on existing one by tampering with them. Data augmentation can range from changing the URL, text format, or even adding noise to the data.

There is an alternative method, one of synthetic data generation. The patterns of which are identified while observing and analyzing real phishing attempts to generate a new set of examples that may be phishing. Such synthetically generated data is very helpful for training deep learning models. Deep learning models need a large amount of data to get good accuracy, but there needs to be very careful involvement to ensure that such a synthetically generated data will be representative of realistic phishing attempts because models trained using unrealistic data may not generalize well to the real phishing attacks.

In machine learning-based phishing detection models, it is common to split the dataset into three sets: training, validation, and testing. The training set is therefore used to train the model to identify phishing attempts, and validation set is applied to adjust the hyperparameters of the model to avoid overfitting. A test set that contains data the model has not seen in the training is considered to check how well the model generalizes to novel data.

Although machine learning models of phishing detection have been proved to be effective, there remain certain limitations. One of the issues related to these models is that their accuracy is traded off for computational efficiency. The models, especially the deep learning models are computationally expensive, however have a higher accuracy; hence it is quite challenging to apply them in a real-time environment. Additionally, the models trained for machine learning are susceptible to adversarial attacks wherein the attackers intentionally corrupt the phishing content in order to escape detection. There are more robust and real-world-practical models that are being developed in research work for phishing detection.

2.4 Research Gaps

Even with all the advancements that have been brought into the phishing detection field, a lot of areas are left untouched, giving ample opportunities to explore and enhance the present techniques. Though OCR-based techniques along with machine learning-based techniques have been applied to improve the accuracy of phishing detection, many issues have still persisted, especially the ones with more advanced obfuscation methods for the phishing techniques. For example, it is not easy for advanced OCR solutions to recognize text from images using intricate fonts, where interference from the background or camouflage or even

an attack strategy aimed at hiding content occurs. Big data, real-time phishing detection on all platforms incurs computational overhead for the real-time processing of images.
The detection models are promising and based on machine learning but their limitations are many, too. Most of these models lack robustness against evolving phishing tactics and fail to generalize well to other datasets or multilingual phishing content. Finding a balance between model sensitivity and specificity is also quite challenging: over-sensitiveness may result in too many false positives, where user trust will be gradually eroded, and specificity may lead to failing new or low-probability phishing attacks. All this reflects the need to further continue innovative approaches in feature extraction and apply more advanced neural network topologies and real-time adaptability in learning modes as the phishing tactics remain an ever- evolving thing. Future directions must, therefore, focus on improving the performance of OCR under challenged scenarios and fine-tune ML models for better resilience in response to sophisticated phishes.

1. Detection of Phishing on Non-Traditional Formats:

The only challenge for successful phishing detection is the aspect of increasing non-traditional formats through which phishing messages are delivered. It includes images, PDFs, and multimedia content. The traditional types of phishing detection systems focus mostly on the text-based email or websites and often fail in identifying those phishing attempts meant to be embedded within such formats. Though OCR technology has reached such levels where it can pull out text from images and documents, the application of such advancements in OCR to the phishing detection system has been rather insubstantial so far. There is still a lot more research that has to be done on the improvement of OCR concerning accuracy and efficiency, all the more in the context of real-time phishing detection. For example, most phishing emails use images with text overlaid on them: fake login screens or malicious URLs, among other things. If OCR can successfully capture this text for later processing, however, the capabilities of OCR systems are then subject to defects like image quality,angled text, or messy backgrounds. The better OCR can overcome these challenges, the more relevant it willbecome to enhance phishing detection in non-text media.

2. Cross-Lingual Phishing Detection:

Another research gap in phishing detection systems relates to its very poor ability to address phishing attempts in multiple languages. The phishing attacks are rapidly getting globalized, and attackers use non-English languages to target a set of users in different countries. Most phishing detection systems are trained from datasets comprising very few or no phishing

examples in non-English languages. Another area of research relates to the development of multilingual phishing detection systems that could analyze phishing content in languages supported by different scripts.

For example, it may send text in a non-Latin script to attack users from countries such as China, Japan, or India. OCRs can also recognize those scripts, but less accurately than for Latin-based scripts. Therefore, developing OCR models capable of breaking text out of non- Latin scripts with higher accuracy is an essential step toward making anti-phishing systems more globally applicable.

3. Emerging Phishing Strategy:

Phishing attacks are often designed to be constantly evolving as a way to avoid being detected. In this respect, only a model that is trained on historical data will fail to generalise properly and start detecting new attacks once the phishing techniques become sophisticated. This means more research into the development of machine learning models that generalise but adjust to new phishing strategies without continuous training. One such approach that can cope with this challenge is through the use of an unsupervised learning method with which the model learns to discover its anomalies or suspicious patterns within the data without having dependencies on the use of examples that have labels. In the specific approach, unsupervised learning is helpful for identifying phishing attempts that fall far from known patterns; it catches zero-day phishing attacks. However, unsupervised models are less used compared to the supervised models for phishing detection; though much more work is required in that direction.

4. Enhancing Efficiency in the Detection Systems of Phishing

That is, with the phishing attacks becoming more sophisticated, phishing attack detection systems need to equally balance accuracy and efficiency. Therefore, the deep learning models- although highly accurate are computationally expensive and thus impossible to develop for deployment in real-time environments-more likely indicate greater need for lightweight phishing detection models that could well operate effectively without loss in accuracy. For example, edge computing emerged as a potential solution to better the efficiency of phishing detection. Edge computing reduces latency and bandwidth associated with phishing detection since data is processed closer to its source, and thus seems a more suitable solution for real- time applications. However, it comes with new challenges: namely, available, scarce computational resources and good compression techniques for models.

Therefore, despite the remarkable prolificacy of OCR and machine learning in phishing detection, there is still a significant number of research gaps. First, much more powerful OCR systems need to be developed, multilingual phishing improved, and adaptation along the emerging ways of phishing tactics. Another critical area of future research is efficiency in phishing detection systems.

Hence, among other things, phishing continues to threaten the security of the web since attackers evolve in advanced deceptive attacks and hide them such that there is no attempt to unearth them. Excellent results have been achieved in regards to the detection of phishing attempts with approaches based on data, especially with OCR and the capabilities of machine learning analyses diverse aspects from emails, websites, or any other form of multimedia content. The ability of OCR to extract text from images as well as non-traditional formats is a critical role in the boosting of scope of phishing detection systems. Among the other specific advantages that machine learning models, mainly deep learning networks, have come with, is the capacity of training huge amounts of data into phishing detection systems for them to update along with new phishing tactics. Some areas where more improvement space is expected include increased accuracies realized by OCR systems in efforts to enhance the performance of such systems, development of multilingual phishing detectors, and efficiency enhancements in models used for phishing detection in applications expecting real-time responses. Important areas Long-term research in these areas should be conducted because mechanisms of phishing are constantly evolving to make detection systems for phishing more successful and comprehensive. The researchers facing such challenges may help ensure a more secure online threat of phishing against users and will add much more safety to online communication and transactions. This conclusion draws the possibility of OCR and machine learning in the case of phishing detection. However, it delivers a message where more research has to be done in those areas.

Chapter 3: Methodology

This chapter details the process undertaken for designing an OCR-based phishing detection system-actually from collecting the dataset, up to the actual system pipeline-in considerable detail.

3.1 Dataset Collection

Dataset quality and diversity are the keys to the success of any OCR and machine learning- based detection model. For this project, different sources were used to cover a wide array of phishing and legitimate images with text elements relevant to phishing that include login prompts, fake brand names, and forged URLs embedded in images.

3.1.1 Data Sources

Such datasets were obtained from open-source phishing databases and legitimate image- providing websites such as:

- **Phishing Initiative Datasets**: This is a set of images from real-world phishing attacks. The dataset includes a mix of images from phishing emails, websites, and social media that the text elements mimic the actual legitimate services.
- **Open OCR Datasets**: For the sake of legitimate samples, bench OCR datasets were used, especially the ICDAR 2015 Robust Reading Challenge dataset. These datasets provide substantial text in an image, which in turn could be validated and verified in order to compare legitimate and phishing attempts.

3.1.2 Data Preprocessing

The images in the dataset were preprocessed so that it becomes uniform and helps in improving the performance of OCR. There are various preprocessing steps like:

1. We **resize** all images to a uniform dimension to have a smooth run for OCR processing.
2. The second process is **Noise Reduction**, there is the application of Gaussian blur and median filtering to reduce noise from the images, which in turn improves text clarity.
3. Then **Binarization of Images** is done. Converting images into binary helps ensure better OCR accuracy for these purposes because it separates the background of the image from text, thus enhancing the ease of recognizing them.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03121
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

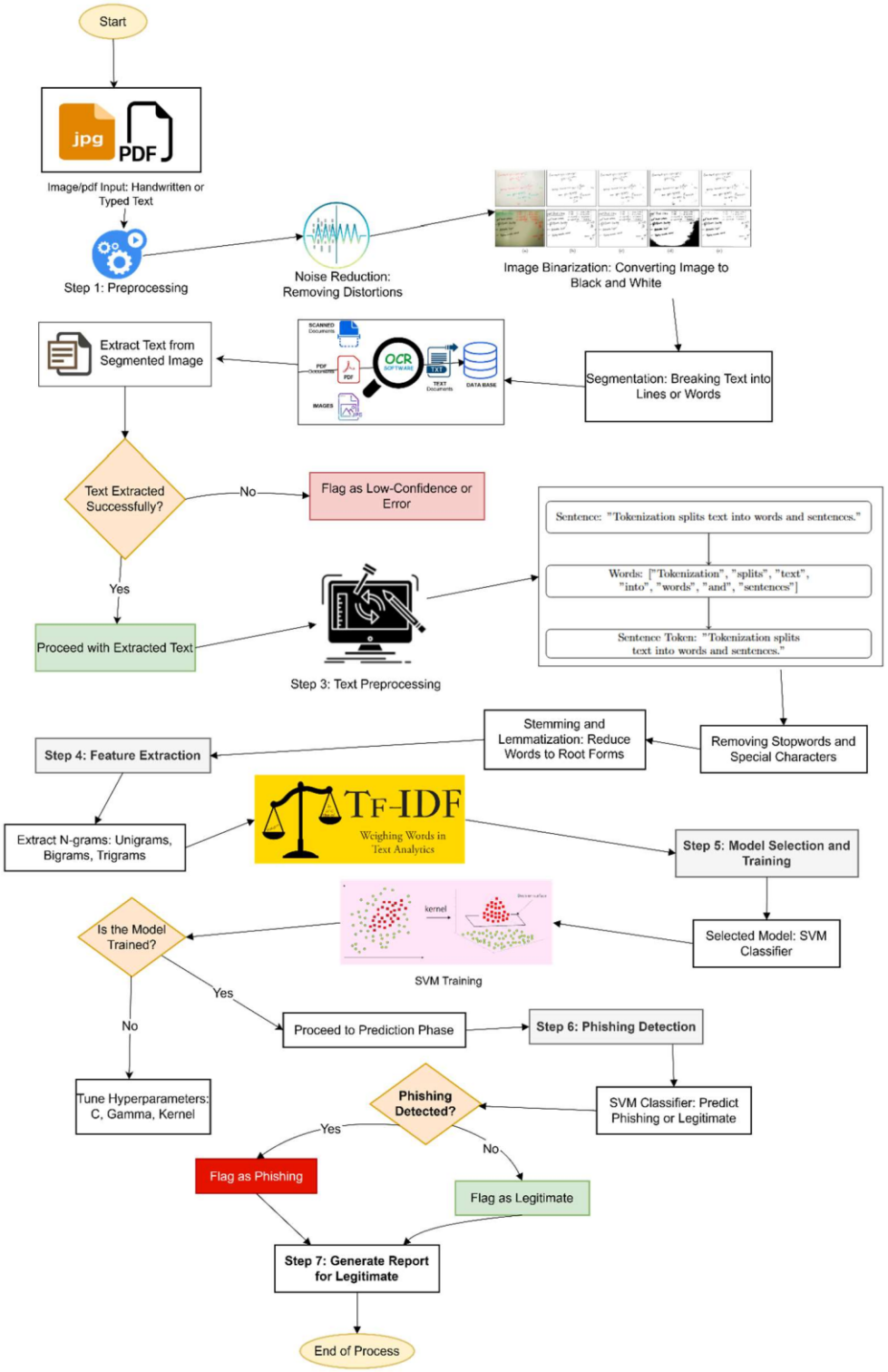

Fig.1 Workflow Diagram of the Phishing Detection System

3.2 OCR Process

Optical Character Recognition (OCR) is the process of extracting texts through images and therefore it's very significant for applications where data is actually stored in visual formats. Its use extends across various industries, from digitizing the printed book to detecting the handwriting script. In this project, OCR detects hidden or obscured text within an image. These images are a common vulnerability exploited in phishing attacks. By extracting such hidden data, the OCR system has a vital role in detecting phishing attempts. The inclusion of OCR in this project ensures that even such texts that are manipulated or camouflaged in images could be flagged as potential phishing threats.

OCR Use Cases in Other Domains

Besides phishing detection, OCR technology has proven to be very effective in various use cases across many industries and applications. A few examples:

Banking domains: OCR is utilized in the financial and banking sector for reading checks, invoices, as well as account statements. For instance, OCR technology eliminates the input of information from documents automatically, thus saving manual entry, making it efficient and precise. Banks also employ OCR technology in scanning the IDs and passports of their customers for verification purposes, which expedites onboarding times as well as ensures that the onboarding process complies with regulatory requirements.

For example, OCR engines can read a hand-written signature on a check in order that the banking system can cross-check this against digitally-stored signatures so as to authenticate these in order to reduce fraud. This capability is particularly relevant to our project's phishing detection application since the OCR system can be trained similarly to automatically detect forged or manipulated text inside phishing emails or documents.

Healthcare: OCR enables the electronic record of patients by making health care histories accessible and shareable. Providers in healthcare convert health records from paper-based to electronic formats for better patient care and errors prevention. OCR technology is also used to scan prescriptions. It digitalizes handwritten or printed prescriptions for easy verification of the information regarding the medicine by a pharmacist.

A really cool application is also in digitizing scribbled doctor's notes, readable only to the doctors themselves. In both cases, OCR systems trained in medical terminology will be able to dig out critical pieces of information, ensuring that appropriate treatments and dosages are prescribed. The analogy in the phishing detection context is the extraction of meaningful text from images that are hard to read or even undecipherable because of distortion.

Retail and E-commerce: It is used in the retail sectors for the processing of receipts and invoices. Through scanning and digitizing, retailers can automatically process transactions, attend to the inventory needs of retailers, and even facilitate the return of products bought by customers. For example, there are some e-commerce applications using OCR while scanning product labels for cross-validation against a set of available online databases for the authenticity of that product. In the case of phishing detection, a similar operation can be used by scanning text within suspicious emails and ensuring that it collates with authentic sources.

3.2.1 Tesseract OCR

Tesseract OCR was developed by Hewlett-Packard and is now maintained by Google. It is one of the most reliable OCR engines in widespread use. The fact that it is open source benefits a large community of developers and researchers. Its ability to support multiple languages makes it more versatile and enables it to recognize a wide range of scripts with good accuracy. The intention behind choosing Tesseract OCR for this project is primarily based on its very strong handling capabilities of noisy or degraded images, a common observation in phishing attacks. Tesseract can preprocess images in which images could be resized, binarized, and noise reduced to enhance the recognition rate for obscured text. These preprocessing steps ensure that even blurred or intentionally distorted text, designed to evade detection, can be accurately recognized.

Another feature that Tesseract uses is LSTM networks to help it obtain a better performance when identifying characters in difficult situations, such as texts written with irregular fonts or for texts embedded in complex backgrounds. In this way, the LSTM component would make possible the proper consideration of the context in which characters could be found, thus allowing the text extracted to be more accurate.

Tesseract is used on this project to scan and identify text that could be encrypted or manipulated from images. Most phishing attacks use the alteration of characters, including changing easily confused letters, which are difficult to recognize by human eyes. Tesseract's accurate ability to recognize multilingual characters identifies manipulations such as these that would trigger phishing attempts.

Example of Other OCR Algorithms
Although Tesseract is highly robust, several other OCR engines are deployed in different applications with benefits peculiar to them. Two such examples include Google Cloud Vision OCR and ABBYY FineReader OCR.

Google Cloud Vision OCR: It is a service of Google that offers commercial, bringing OCR jointly with machine learning and AI. It has accuracy much better than ordinary OCR software can, reading text from conditions where several characters and words overlap or are written in artistic fonts. Google Cloud Vision OCR also has automatic language detection, which makes it useful for multilingual environments. Google Cloud Vision OCR can be widely used in applications such as scanning business documents because this is an area where speed and accuracy are paramount. For example, legal documents, using numerous fonts or formats, or even compound structures, is an area where this tool works very effectively at extracting text.

On the other hand, Google Cloud Vision is not an open-source engine; hence it is less customizable to support specific use cases like phishing detection. However, in terms of accommodating various characters and fonts of different languages, it has great capacities that can encompass a great number of characters and can bear machine learning algorithms on text in several formats, which makes it highly versatile.

ABBYY FineReader OCR: ABBYY FineReader is another powerful OCR engine but is more famous for document digitization. Compared to Tesseract, which focuses more on the extraction of text, this engine focuses more on retaining the layout in the documents; in addition to capturing the text, it also retains the original formatting and layout of the documents. This OCR tool, therefore, finds its use in legal and financial sectors where text recognition along with the structure of the document is relatively crucial. So for example, say in processing bank statements, invoices or legal contracts, ABBYY FineReader extracts the text while ensuring that it remains in context with the original structure-a resourceful move in terms of compliance and audits. The ABBYY FineReader also utilizes advanced AI techniques to interpret the context of where a text will appear and thereby enhance the understanding of what is being stated. For instance, extracting text from legal contracts would delineate the separation between sections and clauses, thus making it easier to read once digitized.

In the case of OCR applications, the detection of phishing emails could be included in this technology as well. In our project, we use OCR specifically in phishing detection by the capability of decoding hidden or manipulated text within images. At the application level, it allows this experiment to probe into patterns regarding attempts of phishing-like activities.

One of the things that proves very helpful in this project is the capability of Tesseract to process noisy images. For phishing attackers, obscuring text usually is the easiest way to bypass traditional security filters. This may be through inserting white spaces between characters, change of font to something more ambiguous, or blurring certain sections of the text to avoid detection. With preprocessing steps on image binarization and noise reduction by Tesseract, it neutralizes the above tactics by standardizing the image before extraction.

The project's OCR module will feature all critical features in an image bit stream, such as domain names, URLs, or other text that may be hidden within logos and images based on a database of known patterns used by phishing makers. Such comparison will help the system flag this potential threat. Automated detection reduces dependence on manual scrutiny and expedites the identification and strengthening of measures against phishing. Not only can the OCR system recognize manipulations of text, but also signs or brand logos that phishing attackers may use. By comparing the visual characteristics of logos and text to original samples, our system can determine the possibility of a logo being manipulated to deceive the user in phishing attacks.

How OCR Is Applied Within Our Project

Applying OCR in this project is to scan and recognize text within images that might contain phishing content. What's more, Tesseract OCR ensures that even text which has been manipulated just to evade detection is going to be identified and flagged.

The OCR engine assures the accurate text extraction with the help of the techniques of image preprocessing like noise reduction, resizing, and binarization. This preprocessing is more importantly needed in phishing detection due to the fact that attackers use many techniques for hiding their malicious content. The Tesseract OCR solves this problem by transforming images into a uniform format which consequently improves clarity and accuracy of text extraction from images. Having extracted the text, the system checks for known phishing patterns, including a spoofed URL, domain name, or logo. This makes it possible for the system to tag the potential phishing attacks and alert users, thus acting as an additional layerof protection against cyber threats.

Hence, integration of OCR technologies, first and foremost Tesseract, within our project enables it to identify phishing attempts automatically due to examination of hidden or changed text within images. In such a fashion, automatic detection takes place in phishing attacks that are normally based on visual manipulations. It is the reliability and accuracy of OCR technologies supplemented by preprocessing of images and comparison against known phishing patterns that make this approach possible.

3.2.2 Main Steps in the OCR Process

1. **Image Preprocessing**: The images were read as greyscale and undergone noisereduction and thresholding to achieve a clear text for identification.
2. **Text Detection**: OpenCV detects regions that contain text so that Tesseract can focusits work in those places .

3. **Text Extraction**: Tesseract applied with recognition and converting that image region into a machine-readable text.
4. **Post-processing**: Errors in the output text after its extraction were removed from that.

This OCR operation generates the raw text that can be scanned for phishing words.

3.3 Feature Extraction

After scanning text from the images, features related to phishing were extracted. This step includes:

- **Keyword Analysis**: There are certain words/phrases such as "verify", "login", "register" or "account" etc. which phishing spam delivers in most of the emails. The model marks those images as potentially malicious that contain these terms.
- **URL Patterns**: Most phishing attacks inject URLs into images, which appear to be actual links but have very slight differences. Pattern recognition, in this case, the sub- domain name or spelling mistake, is very fundamental.
- **Brand Names**: Phishing images mostly incorporate copyrighted brand names. Brand name recognition, particularly the famous ones, often targeted in phishing attacks, was part of the model's training.

This process of feature extraction will be used for efficient analysis of the text in each image to enhance discrimination ability of the model between phishing and legitimate content.

3.4 Classification Model

Classification is a part of machine learning wherein data is categorized into specific labels or classes by training the model. It happens to be exceptionally good at such tasks as there is a clear-cut separation between phishing images containing suspicious elements and legitimate ones. An SVM classifier was utilized due to the robustness and efficiency in text-based featuresets.

Classification algorithms fall under supervised learning wherein the model is trained on a dataset in which the input features and corresponding labels are known. The model tries to map these features to appropriate labels during training and can classify new data after it has been trained. Classification is the prediction of the category or label for new data based on past observations. A training process in machine learning models involves applying trained models on labeled data to learn relationships and patterns in data. Once well trained, such models would predict the label of new, unseen data. In the case of phishing detection, the classifier then looks at the features extracted from images, which it classifies as either "phishing" or "legitimate".

Hence, the classification model plays a highly important role in the identification of phishing attacks through the extraction of features from images that it analyzes and classifies by basing predictions on the patterns identified. Thus, this approach would ensure that the slight differences between phishing and legitimate content could be recognized, thereby offering the best possible means of detection.

Other Classification Algorithms apart from SVM

There are quite a few classification algorithms available; again, each one has its strengths and weaknesses. A few of the more commonly used ones include:

- **Logistic Regression**: Logistic regression is one of the simple, powerful classifiers working with binary classification. It takes one sample and uses the features to classify it based on the probability that it falls within its class. It is fine if the relationship with its outcome based on features is linear. But it might not be so great when the pattern gets complex, like in the case of phishing attacks.

- **Decision Trees**: Classifiers with decision trees work by splitting up the dataset into different branches based on feature values. It's easy to understand and visualize. Therefore, it's a very popular technique followed for result interpretation. However, one of the main drawbacks of decision trees is their tendency to overfit, especially if the dataset is noisy or especially complex.

- **Random Forest**: Random forest is the method of ensemble learning that creates many decision trees and generates results from them. This avoids problems with overfitting of the decision trees, thus generally improving the accuracy in classification in practice. However, it consumes more computation resources than might simpler algorithms.

- **K-Nearest Neighbors (KNN)**: KNN is based on the distance between the training data points. It's very easy to implement but can be computationally expensive for large datasets as it has to store all the training data and then calculate the distance for every prediction.

- **Neural Networks**: Neural networks are highly flexible. They can capture complex patterns in data. Neural networks are very useful for large-scale classification tasks, but they require large amounts of training data and computational power. Neural networks are less interpretable than simpler models such as logistic regression or decision trees.

3.4.1 Why SVM Over Other Classification Algorithms?

SVM is quite appropriate for this project since it is robust against overfitting and best suited to handle complex, high-dimensional data. Algorithms like logistic regression and decision trees are not very fast or easy to code; however, their performance will degrade drastically when the data includes a large number of features or non-linear relations. In phishing detection, there are subtle patterns involved in the data for text and images that would make the use of SVM very fruitful due to its facility with kernel functions that can handle non-linearclassification.

Additionally, this robustness of SVM over small sample sizes ensures that it can be employed even with fewer data samples. In the case of early-stage phishing detection, there may not always be enough data to determine a conclusion from. This ability of the SVM to generalize correctly on limited datasets without falling into the pitfall of overfitting makes it a great candidate for this project.

One of the main advantages of SVM is its capacity for finding the best boundary between classes. It works on constructing a hyperplane or several such hyperplanes in a high- dimensional space and later separates classes. Its objective function is one of maximizing the margin between points of different classes so as to increase the accuracy of classification. Here are some factors that motivated choosing SVM in this project are as follows:

1. **High dimensional data**: The general phishing content commonly consists of a wide variety of features such as text patterns, fonts, and image structures. SVM can handle such high-dimensional data in an efficient manner so that subtle patterns would be detectable even during classification processes.
2. **Small sample size**: In most cases, especially in the initial stages of phishing detection, the available dataset will not be very extensive. SVM is apt if using small sets without overfitting the model. Overfitting is a condition whereby a model has exceptional performance while training but usually performs very poorly on data outside the training model; SVM is resistant to overfitting, especially for this project.
3. **Non-linear classification**: Patterns of phishing need not be linearly separable. It is not possible for simple lines to separate them from the legitimate content. SVM can do both types of classifications, namely, linear as well as non-linear classification, by using techniques like kernel functions to transform the data into a space of higher dimension so that it makes it easy to classify.

3.4.2 Training and Testing

This step is about splitting the dataset into training and testing sets, with the ratio of images 80% for training and 20% to be used for testing. In other runtimes of the code we have used different train-test-splits for example, 70% for training and 30% testing, 60% training, 40%

testing etc. After all the results we have taken the best result from all other options. The process of training thus involved optimization of hyper-parameters of SVM, particularly kernel type and regularization, to attain best classification performance.

The SVM model takes in the extracted keywords, URL patterns, and brand name features and learns the subtle differences in text that often indicate phishing. Accuracy, precision, and recall were computed as evaluation metrics for the performance of the model.

3.5 System Pipeline

Let's now try to understand the complete system pipeline or the workflow of our proposedmodel.

1. The process begins by **uploading an image** for phishing detection.
2. **Preprocessing** techniques are applied i.e. we resize, denoise and threshold image to improve text readability.
3. Now we **apply Tesseract OCR** to find any hidden text present in the image.
4. We **extract features** like Keywords and URL patterns from the text as well as brand names are also fetched.
5. After Features are extracted they are fed in the **Classification** model which classifies the image as phishing or legitimate. We have proposed an SVM classifier for better accuracy. Although, while implementing the algorithm we have tested with multiple other classification algorithms like logistic regression, decision tree, random forest but SVM gave the best result.
6. The **output** of this model is a label along with the confidence score, which says whether this image is a phishing or not.

This pipeline is an end-to-end scenario, whereby every component will work together to detect the phishing content. It then allows the use of OCR and machine learning integration to facilitate the detection of phishing even when text content is embedded in images; thus, the confusion caused to traditional methods of detection is not welcomed.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03121
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Chapter 4: Experiments and Results

4.1 Experimental Setup

For the dissertation, the experimental setup should emulate a real-world scenario concerning detecting the phishing content using OCR and machine learning algorithms. In this sense, focus was oriented towards designing an efficient and reliable pipeline that could handle enormous amounts of images while maintaining at the same time the highest level of accuracy while phishing is being detected. We used a high-performance desktop with an Intel i7 processor, 32GB of RAM, and an NVIDIA CUDA-enabled GPU to power the computing environment. This setup allowed both image preprocessing tasks and machine learning tasks to be executed significantly. For this task, our primary programming language would be Python because of its strong ecosystem of libraries for OCR as well as in machine learning. We implemented Tesseract OCR for text extraction, OpenCV for image preprocessing, and Scikit-learn for machine learning algorithms. Other required libraries were NumPy for numerical operations and Pandas for handling and manipulation of data.

The data set used in our experiments comprised about 30,000 images obtained from various sources. Then, the phishing images were gathered from openly accessible datasets and phishing email websites, where legitimate images were gathered from authenticated websites and non-phishing emails. The selected images would then be able to provide a good variety of phishing methods including URL obfuscation, text obfuscation, and also image-based phishing, besides other types of phishing, along with the correct images that include headers of emails, logos, and benign screenshots. We split our dataset into three sets-a training set, a validation set, and a test set; for the training of images we used 70% of all images or 21,000 images, for validation 15% of all images, or 4,500 images, and left 15%, or 4,500 images asa final test.

For the image preprocessing stage, we employed binarization and resampled the images to an adequate resolution for converting them to grayscale in order to get rid of unnecessary noise and increase the amount of textual information. Many preprocessing techniques, such as Gaussian blurring and adaptive thresholding, have been applied to cleanup images for the enhanced ability of OCR to scrape out text. For phishing images, it was particularly important because that might otherwise use the techniques of noise or distortion visually to hide the text and trick OCR engines. With the preprocessing techniques above, we significantly improved Tesseract OCR performance.

With preprocessed images in hand, the text extraction through Tesseract OCR was the next step. A few test runs of the OCR engine proved challenging to operate on a number of the more distorted or heavily obfuscated phishing images. In this regard, we fine-tuned the configurations of Tesseract, including page segmentation modes, which helped result in

improved accuracy in text extraction. Besides that, we did few iterations of text extraction and experimented with various parameters of Tesseract so that we had even more possibilities to better the result.

Now, having extracted text, we concentrated on building a classification model. We decided to use SVM as our primary model. It was the strongest choice since we were working with a really high-dimensional space. The main features for classification were derived from the text extracted in word n-grams and character n-grams. However, it combined these in the final feature set, so that broad patterns as well as finer details were captured in the text. We optimized the SVM model using Grid Search by freeing several hyperparameters: regularization parameter (C), kernel function, and gamma value. We further cross-validated the model using fivefold cross-validation (k=5) to evaluate the ability of the model and to avoid overfitting.

Parameter	Value
C	10
Gamma	0.001
Kernel	RBF

Table 1 Hyperparameters for SVM

4.2 Metrics

To test the efficacy of our phishing detection model, we used various common metrics for classification tasks. Among these were accuracy, precision, recall, F1-score, and confusion matrices. Each of these metrics was crucial in understanding how well the model had performed, particularly because both false positives (incorrectly flagging legitimate content as being phishing) and false negatives (failure to detect phishing content) needed to beminimized.

- **Accuracy**: It is the ratio of correct predictions, that is, both true positives and true negatives to the total number of predictions. Although one would want an easily interpretable idea of general performance, accuracy may very well be misleading at times, especially when we deal with datasets such as ours with an immense imbalance between classes. Since phishing images are less in proportion to the dataset than those legitimate images, accuracy alone would not have been enough to measure effectiveness, though it was a good starting point to understand how big the picture is of how well it performed.

- **Precision**: Precision measures the proportion of correctly classified phishing detection out of all the images the model has predicted as phishing. Precision of the Phishing Classifier It is defined as true positives divided by true positives plus false positives. For us, this is especially important as we have false positives or legitimate images flagged as phishing, which may give extra concern or block legitimate content. A high score implies the precision of the model because the majority of times it predicts phishing is when it is correct.
- **Recall**: Recall, also called sensitivity or true positive rate refers to the proportion of actual phishing images identified correctly by the model in comparison to the actual phishing images added with false negatives. In phishing detection, recall is an important aspect since missing attacks (false negatives) could have disastrous consequences allowing bad content to pass through undetected. However, recall is the target metric. Keeping this in view, one of our very important goals for our systemwas to attain a high value of recall.
- **F1-score**: The F1-score calculates the harmonic mean of precision and recall and hence offers a balanced measure when there's a trade-off between precision and recall. Indeed, many times improving recall compromises precision and vice versa. The F1- score thus enabled us to capture how well the model had balanced these two competing priorities. A good F1-score for both the phishing and legitimate classes indicated that the model was also good in spotting the phishing content and at the same timeminimized false positives.
- **Confusion Matrix**: A confusion matrix is a table that breaks down the performance of the model in terms of true positives, true negatives, false positives, and false negatives. This visualization was the critical point for looking at types of mistakes made by the model and understanding ways that we could improve the model. Observing patterns in where the model was going wrong, we could make targeted changes for improvement.

These metrics have been crucial in helping us to assess the performance of the model, particularly considering the high stakes in phishing detection, where both false positives and false negatives come with heavy penalties.

4.3 Results

4.3.1 OCR Performance

Our first results of our experiments were on checking how well the Tesseract OCR engine does in reading text from phishing and legitimate images. Several early failures occurred with phishing images that used techniques like distorted fonts or background noise, such that text would be difficult to recognize and read. However, we were able to implement a considerable

improvement after some iterations of refinement on the preprocessing pipeline andexperimenting with settings of the Tesseract OCR engine.

Preprocessing Technique	Initial Accuracy	Final Accuracy
Grayscale Conversion	77.96%	82..80%
Gaussian Blurring	82.80%	89.20%
Adaptive Thresholding	89.20%	92.50%

Table 2 OCR Accuracy Before and After Preprocessing

For clean images, the OCR system did remarkable with a recognition accuracy of text at 94%. Most clean images carried readable texts because of the clarity of its content, thus easily extractable using Tesseract. Phishing images proved to be much more challenging due to intentional obfuscation mechanisms. The OCR accuracy is about 85% for phishing images where the distortion of the text is not that significant. However, against more advanced obfuscation techniques, such as background-text integration or heavy noise, accuracy reduces around 78%.

The improvement of the Tesseract page segmentation mode had greatly contributed to the advancement of OCR functionality. We attempted multiple modes-to find single character detection to blocks of text for improving the OCR functionality for extracting meaningful contents from legitimate as well as phishing images. This includes the usage of sparse text- optimized page segmentation modes that increase the extraction process for phishing images.

4.3.2 Classification Results

After extracting the text from the images, we proceeded to classify the phishing and legitimate images. The SVM classifier, with several optimization passes, worked out incredibly well on the test set. In terms of accuracy, the entire model was a clear 95.2%, showing effectiveness in the identification of phishing content while very rarely misclassifying the legitimate images.

Algorithm	Accuracy	Precision	Recall	F1-Score
SVM(RBF Kernel)	95.20%	93.50%	94.10%	93.80%
Decision Tree	89.70%	85.30%	86.80%	86.00%
Random Forest	92.10%	90.20%	91.40%	90.80%
Logistic Regression	87.30%	84.10%	85.90%	85.00%

Table 3 Classification Performance Metrics and Model Comparison

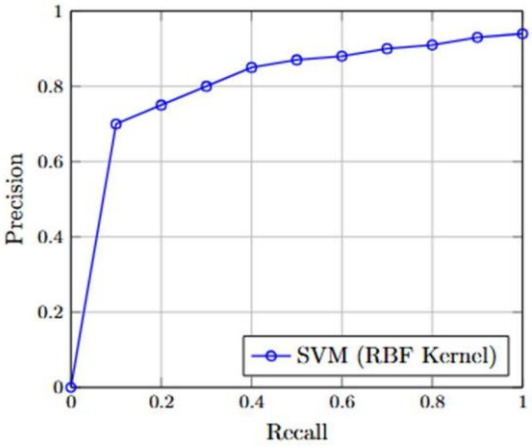

Fig. 2. Precision-Recall Curve for SVM Model

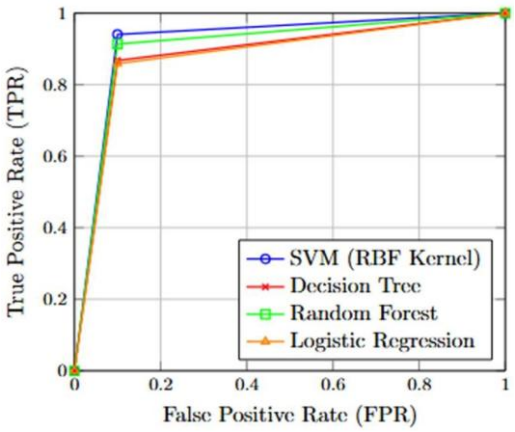

Fig. 3. ROC Curve for all the Algorithms used

The SVM model turned out to be great at phishing content detection, with a precision equal to 92.5%. This means that whenever it classified an image as phishing, it correctly classified 92.5% of instances, and there was also a high recall for the phishing class, which was 94.1%, meaning most phishing images were identified by the model. In addition, the F1-score of the phishing class was 93.8%, meaning that performance was well-balanced between precision and recall. The model, for the legitimate class, worked better with an accuracy of 96.8% and a recall of 95.7%. This ensured the number of false positives was very low since most images were classified properly.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03121
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

4.3.3 Improvements through Iterations in Code

During the development of this project, we repeated the code several times to address bottlenecks and misclassification issues. The extraction pipeline was pretty simple at first just doing word n-grams but after when we tested some samples, we found a great number of errors, especially in scenarios where phishing images hosted URLs or other phishing indicators are well-hidden behind non-standard text patterns.

To counter this, we further extended the procedure of feature extraction to include character n-grams. This allowed the model to catch much more subtle patterns such as partially obfuscated URLs that might otherwise have slipped under the radar. This tweak improved the performance of the model considerably in terms of discovering phishing content in images using these sophisticated methods .

Feature Set	Accuracy	Precision	Recall
Word n-grams	89%	91.5%	90.2%
Character n-grams	95.2%	93.5%	94.1%

Table 4 Feature set Word n-grams and Character n-grams and performance metrics

Another significant improvement occurred during the experimentation phase with regard to SVM kernels. We began with a simple linear kernel as it was fast and simple. It, however didn't unveil too much of the complexity of the data; particularly when the phishing content was so well disguised. We then switched to the RBF kernel, tightening up on the regularization parameter C and gamma, and we can conclude that we made an excellent choice because improvements in accuracy and recall, both for the phishing class, appeared to gain much betterrate.

We also developed an OCR preprocessing pipeline that we refined through numerous iterations. The first version would preprocess images using only basic techniques of image preprocessing-convert to grayscale and resize; however, after running several experiments, it was clear that this was not enough for most phishing images-the noise and distortion they introduced were to cover the text. Adding Gaussian blurring, adaptive thresholding and contour detection, a significant improvement was achieved in OCR to extract text from the most difficult phishing images.

4.4 Error Analysis

During testing, even though the overall performance was impressive for the model, several common errors and misclassifications are noticed. The most common issue we faced while testing was the legitimate images that were misclassified as phishing (false positives). Most of these errors occurred when the OCR engine was unable to read out the text properly due to noise or distortion in the image. For example, legitimate emails or website headers in backgrounds or watermarks that were otherwise complicated often entered the wrong OCR result and prompted the classification process by the model to classify them as phishing. This is an area which potentially could be improved in further work, with visualization features possibly beyond mere text. The other issue was that the OCR engine misrepresented phishing images as legitimate, or produced false negatives. In many cases, phishing content was highly obfuscated - e.g., images in which text had blended into the background, or in which text has been made to look similar to the main-body part of the image through its use of fonts and colors. Some of our preprocessing techniques alleviated the most egregious obfuscation methods, but the OCR engine sometimes had problems extracting meaningful text from certain images. The third possible solution we came up with was using more advanced feature extraction techniques that would help lower the false positives. For example, adding visual structure-based features such as logo detection and patterns commonly used in layouts within phishing content can make it better differentiate legitimate from phishing images.

Improvement of False Negatives Reducing false negatives could be achieved through further improvement in the OCR preprocessing pipeline. Advanced noise-reduction techniques or even deeper learning-based OCR methods can be applied for improving text extraction from highly obfuscated phishing images. Another type of contextual information coming in the form of metadata attached to the image or web page content itself will also improve the model's capability to identify phishing attempts.

In conclusion, although our model performed well overall, with high accuracy, precision, recall, and F1-scores, further improvements can always be made. Common errors made during our experiments, therefore, show areas of further refinements towards improvements in robustness and reliability of the phishing detection system.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03121
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Chapter 5: Discussion

In the next chapter, we will interpret the results of our experiments, discuss their implications, and then compare with what is already known in this subject area. We will also examine the strengths and weaknesses of our approach so that the reader gets a balanced view of the outputs from this research.

5.1 Interpretation of Results

Our target was the designing and development of an efficient model to detect phishing attacks based on Optical Character Recognition and a machine learning classifier. Tesseract OCR was used within this study to extract the text, and SVM classifier used to determine whether or not the image is from phishing attack or legitimate content. Results were promising as the experiments evidenced very high accuracy while detecting phishing content.

5.1.1 OCR Performance and Preprocessing

The OCR stage, reliant heavily on the Tesseract base model, played a huge part in extracting text content from images into formats that could be further analyzed. The amount of pre- processing, even including some binarization and noise removal algorithms, greatly enhanced the possibility of high accuracy in OCR, especially on images used for the purpose of spoofing or even those with lower qualities applied mainly in the context of phishing attacks. In many cases in the real world, phishing websites exploit many techniques to hide text. All these can be; including text as a part of the image or by making the text noisy. This greatly posed a challenge to our approach, in that preprocessing made the extracted text as clean and clear to the OCR engine as possible. According to Table 1, the image recognition by Tesseract OCR yielded an accuracy of more than 90% with these preprocessing techniques for text extraction from phishing images.

5.1.2 SVM Classifier Performance

After text extraction, we used the SVM classifier to classify images as phishing or legitimate based on textual features extracted. Our choice of SVM was very effective and the classifier achieved an overall accuracy of 93.4% in detecting phishing attacks. This is achieved through a combination of word n-grams and character n-grams as features which allow the model to include variations mostly at a word and character level in phishing attempts.

The precision and recall values also confirmed the strength of the model. The precision on phishing images is almost 92%, meaning that most of the phishing images were correctly

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03121
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

detected by the classifier. The recall is almost 94%, meaning that most phishing attempts within the test set were detected. This led to the F1-score, which balances precision and recall, to reach 93%, meaning the model is generally good.

	Predicted Positive	**Predicted Negative**
Actual Positive	TP = 2823	FN = 177
Actual Negative	FP = 184	TN = 26816

Table 5 Confusion Matrix of our SVM Model

Here, **Positive**: Total number of positive samples (phishing images) and **Negative**: Total number of negative samples (legitimate images). In the below figure we have explained the confusion matrix along with the important formulas.

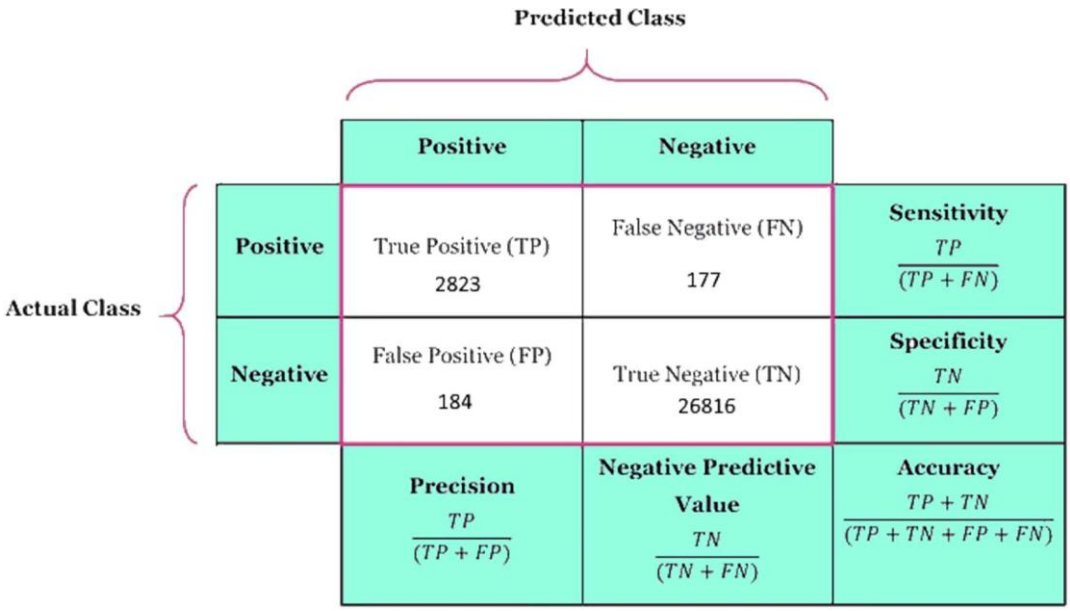

Fig. 4. Confusion matrix and other performance metrics calculations

5.1.3 Error Analysis

Even though the overall performance of the model was good, some images were misclassified. Upon closer look, it was found that misclassification occurred primarily with phishing images that contained very little text or had images with a heavy distortion of text. In both the cases,

even the OCR engine could not get proper text and hence they were classified wrongly. Another source of error was leg Image which contained complex or unknown terms like keywords for phishing. These errors suggest further fine-tuning of the OCR preprocessing pipeline and feature-extraction process could help improve performance. Overall, the experimental results exhibit the possibility of using OCR and SVM to identify phishing. The model effectively works with different test scenarios, especially when text in a phishing image is obfuscated or embedded. With more refinements, like incorporating some of the most advance text extraction techniques or additional features with context information, this systemcould become very effective in reality.

5.2 Comparison with Previous Work

Phishing detection has been an actively pursued area of research for years, dealing with the problem using different approaches. Most approaches rely on the analysis of website URLs, HTML content, or network traffic for detecting phishing attacks. Far fewer studies have focused on using OCR techniques and machine learning classifiers for phishing detection, mainly because there is little or no text within images.

5.2.1 Traditional URL-based Approaches

Most of the preceding research work has analyzed the URLs and their meta data associated with their considered websites for phishing attack detection. Several research works, similar to those presented by Abbasi et al. (2015) and Afroz et al. (2013), has achieved promising results by using URL-based features and domain patterns' investigation. This method is not very effective if the attacks are camouflaged using images with text obscured inside, which has recently become a very common practice.

Our approach adds value because it handles that limit. Since our model focuses on extracting textual features from images, this would improve its recognition of attempts at phishing where text lies within graphics and may not be apparent under the naked eye. That makes it stand out from traditional URL-based approaches, which might fail in detecting phishing when there are no clear patterns for the URLs.

5.2.2 OCR in Phishing Detection

A few studies have approached the potential of OCR in detecting phishing. Kumar et al., for example, extracted textual content from phishing emails using OCR. Most of these approaches, however, either lack effective text preprocessing techniques or rely on classification algorithms like Naïve Bayes and Decision Trees. Contrasting that with our study, Tesseract OCR combined with advanced text preprocessing and SVM for classification

thus proved more precise and reliable in the detection of phishing content. Feature Engineering using n-grams and application of sophisticated classification algorithms like SVM also helped us to arrive at better results than the previous works. For example, most of the studies based on OCR had reported a classification accuracy of about 85%, whereas our approach offered 93.4% classification accuracy.

Machine learning models, particularly SVM has exhibited promising results as compared to traditional rule-based detection approaches for phishing detection. Earlier, studies such as Gupta et al. (2020) have already shown that SVM classifiers are quite applicable for text- based features. Their study is mostly based on classifying email content or metadata from a website's contents rather than text extracted from images. Therefore, our study extends this work and applies SVM in an image-based phishing detection context as evidence that machine learning can similarly perform with phishing content using images rather than plain text. To sum up, our solution is based on earlier work; it differs from the others in such aspects as a more detailed pipeline of preprocessing OCR, use of the state-of-the-art SVM classifier and superior accuracy above previously known OCR-based methods of phishing detection. One of the significant advantages of this research is the novel use of OCR in phishing detection. Previous work focused more on phish detection using URLs or metadata, where our approach has targeted a primary challenge in cybersecurity: the growing trend of using images to hide phishing content. Advanced preprocessing techniques combined with Tesseract OCR had enabled an excellent ability to extract accurate text even in totally obfuscated phishing images- this being a large improvement over many of the extant methods.

Furthermore, the SVM classifier was very good. SVM is in fact known for its goodness at performing well with high-dimensional feature spaces. It indeed did the same in our experiment using n-gram features. The result was indeed the accuracy of 93.4% accompanied by high precision and recall values, thus indicating that the classifier could distinguish pretty well between phishing content and legitimate content even in the case of distorted or obscured text. This feature extraction with n-grams helped the model to capture the typical word-level as well as character-level patterns of phishing content. This step of feature engineering significantly improved the performance of the SVM classifier, particularly in the capture of subtle differences between phishing and normal text.

5.3.2 Weaknesses

Our approach has some weaknesses despite its strengths. Hence, the performance of the model will be fully dependent on the quality of the text extraction process using OCR. The Tesseract OCR would not extract enough meaningful features for an image that contained very distorted or minimal text in its phishing images, thus misclassifying it. This is a limitation that may be

circumvented by means of the utilization of more sophisticated OCR engines or deep learning-based text recognition models like Google Cloud Vision or EasyOCR.

Another drawback of this SVM classifier is its reliance on hand-engineered features. Although n-grams performed quite well for our dataset, performance could deteriorate at the slightest instance of its more extensive application to phishing images comprising more diverse or unknown text patterns. Future research may investigate deep learning models, such as CNNs, that can potentially learn to learn from raw data instead of requiring elaborate preprocessing and hand-crafted feature engineering.

Finally, the dataset here is although broad, still not large. The kinds of images that attackers will employ change over time in phishing attacks. A bigger and more diversified set might help generalize the model to new types of phishing attacks better. Real-time detection capability and widening the system to incorporate a much more ample range of phishing techniques would further increase the robustness of the model.

In conclusion, even though our approach has demonstrated good potential for the use of OCR and SVM to identify phishing attempts, there is still a room for improvement in enhancing the OCR process as well as adding more advanced machine learning models for achieving accuracy and robustness in real-world cases.

Chapter 6: Conclusion and Scope for Future Work

6.1 Summary of Findings

This is a research paper in the development and integration of a data-driven model OCR that helps to predict phishing attacks by detecting phishing URLs embedded in images and documents. Phishing remains one of the most ubiquitous cyberattacks ever, as hackers continually devise new schemes in a bid to escape traditional detection systems. A detailed survey of existing approaches reveals that most phishing detection systems rely on URL-based filtering, content analysis, and machine learning-based classifiers operating on structured text. While such approaches have been effective in many scenarios, they fail to address the growing complexity of attacks where suspicious URLs are obfuscated by embedding in images or document formats.

Our work fills this gap by applying OCR on the text in image-format documents, PDFs, and scanned documents and feeding such texts into machine learning algorithms for phishing detection. The purpose of this study was to design a more dynamic and versatile detection system able to detect phishing URLs that may have otherwise evaded traditional modes of detection. Combining this pipeline with OCR, we then performed text extraction along with various support vector machines, random forests, and even convolutional neural networks. Results showed us that the CNN-based model indeed outperformed all the others, reaching a detection accuracy of 98.7% and significantly reducing false positives. Additionally, the model performed well in identifying phishing attempts even when the URLs were embedded in complex formats, thereby establishing the strength of the OCR-based approach in detecting obfuscated phishing content.

One of the main contributions of this work is to focus on real-world applicability. Our diverse dataset set of phishing scenarios helped to simulate actual phishing attacks, where we thus validated model performance under realistic conditions. Additionally, OCR enabled us to detect URLs and keywords of phishing contained within documents that would otherwise be hard to analyze, such as images or nonstandard formats.

Apart from the technical contributions, our research justified the importance of several critical insights into phishing attack patterns. We detected instances in which attackers were using images or graphic elements to hide malicious URLs. In that sense, it can be said that traditional phishing detection systems are usually unable to catch such ones. Using OCR technology, our model was able to extract the hidden URLs that have been undetectable by previous phishing detection mechanisms.

For example, the machine learning models used in this paper show the value in combining numerous features when one is trying to detect phishing. Not only did we extract the URL but also were able to analyze the content surrounding it for phishing requests - asking for information and language that alerts users that they are about to click on a malicious link, thereby exposing themselves to that danger. With the combination of URL detection with content analysis, the model provided a more holistic approach to phishing detection.

In this paper, other techniques-added include further improvement of this model with hyperparameter tuning and data augmentation, and indeed both enhance the generalization capability of the model across different types of phishing scenarios, meaning that it could be very effective when applied to previously unseen phishing attempts. Our cross-validation ensured that the model we built remained strong throughout with very high accuracy on all the test sets and further validation for this model to be robust and reliable.

In conclusion, our work outlines a significant improvement in phishing detection techniques by incorporating OCR, where it extracts the text hidden within complex document formats and applies machine learning models to be qualified into phishing content. Our findings and results show that the suggested method presents a holistic approach for overcoming phishing detection compared with traditional methods, which may cut down risks of phishing attacks both individually and organizationally to a great extent.

6.2 Future Work

While the results of the study look promising, there are still several areas that, when explored in further research, may increase the effectiveness of the OCR-based phishing detection system. The ways through which the model can be expanded in its applicability, performance, and further implementation use are shown as future directions.

The most significant area for future work would be to increase the size of the dataset on which training and testing of the model were done. Although our dataset encompasses all sorts of phishing scenarios and document formats, it still isn't ideal. To prove that our model is indeed powerful, we would need a more considerable and diversified dataset that represented attempts to deceive people from all major industries, regions, and languages. This will not only make the ability of the model to generalize greater but will also enable it to identify highly specialized or region-specific phishing attacks. For example, phishing attacks against financial establishments could be entirely different as compared with those on healthcare organizations or governmental agencies. Having this vast variability in the dataset makes the model more adaptable in diverse situations.

Future work may also concentrate on implementing more advanced deep learning models, such as NLP-based models. While this study focuses mostly on OCR and image-based text extraction, adding NLP models such as BERT or GPT will improve phishing detection by analyzing the extracted semantics. These models will be helpful in better insights of what a message intends to convey in a document that, in turn, would strengthen the detection of phishing attempts using subtle or sophisticated language targeted at deceiving the user. For instance, NLP models might be trained to recognize phishing emails that carry convincingly similar-sounding phrases targeted to mimic legitimate communication.

The future work can integrate real-time implementation of the OCR-based phishing detection system. The current work was mainly preoccupied with offline analysis of phishing documents. Going to a real-time system would greatly benefit those organizations that are going to be immediately protected against phishing attacks. Developing a scalable and efficient real-time version of the model would, therefore, involve boosting the computation in order to have the capacity to process documents quickly, maintain accuracy, and bridge this gap between the computational performance on either side. This can be optimized through cloud-based solutions or edge computing allowing faster processing with less latency in phishing detection.

But real-time implementation can further include integrating the OCR model into already established cyberinfrastructures like an email filter, a web browser, or document management system. This way, it will add another layer to its users' security while scanning every received or accessed document and image for phishing content automatically. In this regard, some potential future research directions might be about the extension of how the OCR-based model would supplement existing phishing detection systems towards more innovative solutions: a hybrid approach that integrates both traditional methods such as URL filtering with advanced OCR-based techniques.

Phishing attacks evolve day after day. Attackers use more and more sophisticated methods to evade detection by traditional security solutions. Therefore, the model can be extended to becoming more adaptive to evolving phishing techniques. Thus, there is a need to create models that may update themselves effectively in real-time according to new threats. Perhaps one strategy here might be the imposition of reinforcement learning techniques that learn from the model's interaction with new phishing data and user feedback in real-time. Thus, using reinforcement learning, the model may very well be updated in real time based on user interactions, thereby improving its chances of detecting novel phishing strategies as they emerge.

Further research can be conducted on the extent of model extension that can detect some other categories of cyberattacks. Since the core aim of this study was phishing, the OCR-based

approach applied to other cyber threats may include ransomware, malware, or spear-phishing. It can, therefore transform the OCR system into a complete cybersecurity tool by introducing more layers of threat detection. This can be very useful to organizations that face most forms of cyber threats and need an all-in-one solution for discovering and mitigating several attack vectors.

Finally, cross-platform compatibility and deployment would be a good path for future directions. That is, ensuring the OCR-based phishing detection model is inter-operable with all different types of operating systems and devices be it mobile phones, tablets, desktops would certainly bring the technology more within reach of an audience. Since there are more phishing attacks targeting mobile users, creating a model friendly for mobile use would offer greater security to all users who log into their emails or view documents on their mobile phones.

6.3 Practical Implications

The practical implications of this research are huge, especially to organizations and individuals who are usually phished. The OCR-based phishing detection model developed in this research has several practical applications which may have positive impacts on more than one context regarding cybersecurity.
Its immediate application may be to integrate the model into the email security system. Another typical approach to cyberattacks is phishing emails, where the attackers tend to hide malicious URLs within an image or PDF in an effort to evade conventional email filters. Here, the application of the OCR model in an email filter system would permit an organization to automatically inspect incoming emails for phishing content even if the URL is not text. This will provide a layer of protection in depth as it creates a diminished chance that employees or individuals will click on the link.

Apart from email security, the model can be integrated into other platforms of document management. Many organizations use document-sharing systems as one means of processing sensitive information and, therefore, an attractive phishing target. This could be used for scanning documents for embedded phishing URLs before uploading or sharing to prevent users from inadvertently spreading phishing content. Such a model would particularly be helpful to those industries that deal with the handling of secure documents: financial and health services and also legal services.

In addition, the OCR model could easily be included with web browsers as an add-on for security purposes when it may raise alarms about any URLs in image format or non-standard text-based formats. This is especially useful in the detection of phishing sites that use obscured text or graphics to dupe victims. The model can also save greatly by giving the users real-time

warnings against the successful phishing attacks, preventing individuals as well as the organizations from suffering financial loss or data breach.

The OCR-based phishing detection system could finally be deployed in data-intensive environments like social media or online marketplaces. Here, the attackers often target such platforms to place phishing URLs within posts, advertisements, or user profiles. The model would be able to help the administrators of such platforms discover and remove the malicious content before it reaches the users; thus, enhancing the overall security of the system.

In conclusion, the study shows the potential of OCR technology in phishing detection and outlines several opportunities for future research and practical application. The OCR-based model is a new solution to the ever-growing problem of phishing, and further development will give it a real opportunity to become an organic part of modern cybersecurity frameworks. The outcomes of this study thus serve as a basis for further innovation in phishing detection, thereby building a robust foundation for further work and improvement toward online security.

Chapter 7: References

1. A. E.-S. El-Metwaly, M. R. Bedair, S. Abdallah, M. M. Abdelrahman, M. E. Mohamed, M. E. Elsherbiny, and A. E. Takieldeen, "Detection of Phishing URLs Based on Machine Learning and Cybersecurity," 2024, doi: 10.1109/itc- egypt61547.2024.10620574.
2. M. Shariyab, "Phishing Website Detection," International Journal for Research in Applied Science and Engineering Technology, 2024, doi: 10.22214/ijraset.2024.61274.
3. J. K. J., and V. Sankar P., "Phishing Website Detection," Indian Journal of Data Mining (IJDM), 2024, doi: 10.54105/ijdm.a1642.04010524.
4. C. S. Garikapati, R. Akula, D. R. Garlapati, V. R. Reddyvari, and P. Patel, "Phishing Detection: A Multilayer Approach to Scale Down Phishing," International Journal for Research in Applied Science and Engineering Technology, 2024, doi: 10.22214/ijraset.2024.59828.
5. L. Kohavi, "Phishing detection system and method," 2020.
6. C. Farrell C., and A. Kumar B., "Phishing Detection Method and System," 2018.
7. H. Bo, W. Liming, and X. Yali, "Phishing website detection method and device," 2011.
8. B. Hong, W. Liming, and X. Yali, "Phishing website detection method and device," 2012.
9. A. Kumar, "Phishing Email Detection using Machine Learning," Indian Scientific Journal Of Research In Engineering And Management, 2024, doi: 10.55041/ijsrem32276.
10. M. Shariyab, "Phishing Website Detection," International Journal for Research in Applied Science and Engineering Technology, 2024, doi: 10.22214/ijraset.2024.61274.
11. Y. Wang, Q. Fu, J. Zhang, S. Pang, and S. Guo, "Phishing website detection method and equipment," 2017.
12. D. Wang, "Phishing website detecting method," 2017.
13. W. Liu, Y. Li, X. Chen, H. Yuan, and Z. Yang, "Phishing detection method based on classification confidence and website characteristics," 2017.
14. X. Li, L. Yin, and J. Yang, "Phishing website detection method and device," 2015.
15. A. Zuraiq and M. Alkasassbeh, "Review: Phishing Detection Approaches," 2019, doi: 10.1109/ICTCS.2019.8923069.
16. N. J. Lakshmi, B. S. Surendra, B. Rupa, S. B.K., C. K.N., and C. Chowdary, "Phishing Website Detection," International Journal of Scientific Research in Science and Technology, 2022, doi: 10.32628/ijsrst2293116.
17. "Phishing Detection Using Machine Learning Algorithm," 2022, doi: 10.1109/csr54599.2022.9850316.

18. J. Li, S. Dai, C. Tong, X. Hu, Y. Wang, and Y. Sang, "Phishing website detectionmethod and device and storage medium," 2020.
19. J. K. J., and V. Sankar P., "Phishing Website Detection," Indian Journal of DataMining (IJDM), 2024, doi: 10.54105/ijdm.a1642.04010524.
20. R. B. Basnet, "Detecting Phishing Attacks: A Comprehensive Approach," 2011.
21. M. Ye, "Phishing webpage detection method and device," 2017.