**ABSTRACT:**

*The prevalence of cybercrime is directly proportional to the growth in the number of people using*

*the internet. There has been evidence of phishing's extensive usage since its beginning, and it is*

*now the most successful cyberattack vector. According to our findings, phishing is the most*

*prevalent kind of cyberattack, and it employs several techniques to deceive its targets. Phishing*

*attacks using malicious URLs, emails, and websites are rather common. Phishing emails continue*

*to pose significant cybersecurity threats, necessitating robust and intelligent detection*

*mechanisms. Using a large-scale phishing email dataset, this research investigates the creation*

*and assessment of sophisticated ML models for detecting phishing emails. Several ML models*

*were used, including Logistic Regression, XGBoost, Decision Tree, and SVM. The best answer was suggested by using the Logistic Regression model. Featuring an F1-score 99.24, a recall 99.55%, a precision 99.61%, and an accuracy 99.55%, the Logistic Regression model accomplished remarkable results. Comparative analysis against existing models, including Naïve Bayes, RNN, and SVM, highlighted Logistic Regression superior efficacy in detecting phishing emails. Furthermore, training and testing evaluations demonstrated minimal overfitting and consistent generalisation. This study underscores the potential of Logistic Regression in real-time phishing email detection systems, offering a reliable solution to mitigate phishing threats effectively.*

**CHAPTER 1**

**Introduction**

The ability to effectively communicate is crucial in today's society. People often utilise email as a means of more rapid and effective communication. Email is become an unavoidable aspect of daily

life. The communication process is now simpler, quicker, and less expensive, thanks to email. The popularity of it has grown . Forrester Research shows that 20 per cent of consumers refuse to open emails or attachments, even if their email looks legitimate, due to their loss of trust. Phishing emails are a severe issue for which there is yet no ideal remedy. No model is flawless, even if there are several nice ones available today. Enhancing phishing detection models is essential because phishing emails may be quite deceptive. The advent of file-encrypting ransomware has made phishing attempts far more deadly, yet these misleading emails still manage to inflict millions of dollars in harm.

The issue of phishing emails has grown in recent years. According to experts, " phishing assaults

persist in endangering not only the economy and global security but also companies and

consumers"[3]. This highlights the need for robust defences against phishing attempts that originate

in email messages to protect vital services like banking.

The importance of phishing email detection to user security has lately garnered considerable

attention. As a result, several techniques have been developed to detect phishing emails. These

methods range from content-based screening to communication-oriented approaches, including

authentication protocols, white-listing, and blacklisting [4]. Blacklisting and whitelisting are not widely employed since they have not been shown to be effective enough in other areas. At the same time, content-based phishing filters are quite effective and have seen extensive application[5]. To combat this, researchers have been focussing on content-based procedures, such as developing ML and data mining algorithms that leverage the contents and headers of emails [6].

As one can understand, the detection of phishing emails is crucial to combat these attacks. Phishing email detection is an active research area for more than a decade. However, with the expansion of phishing emails, the eﬀectiveness of earlier detection approaches, which relied mostly on ﬁltering techniques, like heuristic and black listing, is poor [17].

As a next step, researchers attempted to exploit Machine Learning (ML) methods that focus on the emails’ contents,such as the email headers, domain, hyperlinks, and word lists to detect phishing emails; nevertheless, the email’s contents can be forged leading to false conclusions [9]. Currently, the evolution of Natural Language Processing (NLP) and, more speciﬁcally, word embedding techniques have contributed to the development of robust phishing email detection approaches that emphasize the morphology and semantics of the emails’ text [10]. Recent works treat the phishing email detection problem as a text classiﬁcation task, namely, they take into account only the emails’ text and apply NLP methods to handle the textual features [7].

The Term Frequency Inverse Document Frequency (TF-IDF) [35] is a well-known method to measure the signiﬁcance of a word in a document. In the last couple of years it has been the most used NLP technique in the phishing email detection ﬁeld, where it was deployed as a weighting factor of the words that appear in the email corpus.Word2Vec [28] is a popular method for the creation of word embeddings, namely vector representations of a word, which has seen a few applications in the phishing email detection for the identiﬁcation of word associations between diﬀerent emails of an email corpus [33].

Furthermore, recent advances in ML, such as the emergence of a new language model known as Bidirectional Encoder Representations from Transformers (BERT) , have revealed promising results in an wide range of classiﬁcation problems.

1.1. 1 Definition of phishing

Phishing is a word that has thousands of references in science journals, a lot of newspaper coverage, and a lot of scrutiny from organizations like banks and law enforcement agencies. This, though, raises the question of what exactly phishing is. The phenomenon of phishing is specifically specified in some publications; in others, it is explained by an illustration, while others presume that the reader already knows what phishing is. Many scholars have proposed their own definition of phishing, resulting in a plethora of meanings in the scientific literature. The literature does not provide a clear description of phishing attacks, which is due to the fact that the phishing issue is broad and encompasses a variety of scenarios. According to PhishTank, for instance: “Phishing is a fraudulent attempt, usually made through email, to steal your personal information” [16]. The definition of PhishTank remains true in a variety of situations that approximately encompass the bulk of phishing attacks (although no accurate studies have been made to reliably quantify this). Nevertheless, the term confines phishing attacks to the theft of personal data, which is not always the case.

1.1.2 Phishing history

As per the APWG, the word phishing was introduced in 1996 as a result of social engineering attacks by web scammers against America On-line (AOL) accounts [23]–[25]. Fishers (i.e. attackers) use traps (i.e. socially engineered messages) to catch fish (e.g. steal personal information of victims) [22]. The origins of the phi substitution of the character f in fishing can be traced back to one of the early ways of hacking, known as Phone Phreaking, which targeted telecommunications networks [22]. As a consequence, ph has become a popular hacking character to replace f. According to the APWG, hackers were using stolen accounts as a form of money to swap hacking codes in exchange for the stolen accounts. Phishing attacks began with the theft of AOL accounts and evolved to include more lucrative targets such as online banking and e-commerce services [22]. Phishing attacks now threaten not just system end-users but also technical staff at service providers, and they use advanced tactics like Man-in-the-Browser (MitB) attacks.

1.2 Problem Statement

Recent surveys point out that a major drawback in the phishing email detection ﬁeld is that previous researches did not consider the advancement of phishing email attacks (i.e., they are using email samples that came from old sources). A second motivation point is the fact that the eﬃcacy of an ML phishing detection method that focuses on the email’s body text using NLP heavily relies on the cooperation of the NLP method with the ML algorithm.

However, previous works have not considered deploying several combinations of NLP techniques with ML algorithms (hereafter, the combination of an NLP method with an ML algorithm is referred to as NLP/ML) to identify the most powerful pair (i.e., the NLP/ML pair that achieves the best performance). Overall, we argue that new eﬃcient detection technologies are needed to limit the ongoing threat of phishing email attacks.

1.3 Aims and Objectives

1.3.1 Aims

The main aim of this study is to develop accurate phishing emails detection method using ML phishing that focuses on the email’s body text using NLP, which heavily relies on the cooperation of the NLP method with the ML algorithms used.

1.3.2 Objectives

Through achieving these objectives, this research aims to provide a comprehensive

solution for phishing emails detection method using ML phishing that focuses on the email’s body text using NLP:

* Collect the phishing email dataset, combining phishing and legitimate email data, to establish a reliable foundation for classification.
* Designed an effective data preprocessing, including tokenisation, removal of stop words, punctuation, and irrelevant features, for improved model performance.
* Develop Advanced Phising Detector models that enhance accuracy and efficiency through Natural Language Processing.
* Conducted detailed comparisons of models using metrics like F1-score, precision, recall, and accuracy to identify the most effective detection model.
* Creating an Intuitive User Interface to help non IT personnels to use the System .

1.4 Scope and Significance of Study :

1.4.1 Scope

The key research areas in phishing email detection include NLP, ML algorithms and optimisations techniques used in phishing detection email, text features in phishing email, datasets and resources used in phishing email, and evaluation criteria.

As previously stated, a phishing attack begins with an email sent to an online customer. This email contains a fraudulent link that redirects the user to a fake website, which is cloned by the attacker to seem exactly like the original website on which it is based.

1.4.2 Significance Of Study

The modern world faces several threats including the significant one of phishing emails, which cause huge financial losses. The preventive methods commonly used today have not proven effective against this threat despite their constant revision. On the other hand, phishing emails have been increasing at unprecedented levels in recent years. To counter this threat of phishing emails, more advanced phishing detection technology is necessary.

Anti- phishing technology developed on the source code features is quite slow in terms of the classification of phishing emails given its dependence on third-party services and scraping of the email content. Many ML methods have been adopted to identify phishing emails, but these cannot effectively detect new phishing scams, which needs significant manual feature engineering. We present a survey analysis of actual phishing email identification works from various perspectives. This survey is unique in the sense that it relates works to their openly available tools and resources.

The analysis of the presented works revealed that not much work had been performed on phishing email detection using NLP techniques. Therefore, many open issues are associated with this phishing email detection. An evolving research area is illustrated by the phishing email detection. Hence, the researchers are in dire need to perform more research efforts to assess DL techniques in the phishing email detection domain

**CHAPTER 2**

2.0 LITERATURE REVIEW

2.1 Related Work

The majority of phishing email detection approaches in the literature process the email’s text to identify text-based features and deploy ML and Neural Network (NN) methods to distinguish phish- ing from non phising emails. The application of both NLP and ML/NN for the extraction of informative features from the emails’ text and the classiﬁcation of emails respectively has played an important role in the phishing email detection. Previous works in this area have employed contextual [43], semantic [42], and syntactic [31] features from the emails’ text. A recent work that has the same ground as ours is proposed in [40].

The authors focused on the emails’ text to distinguish phishing from non phising emails. To do so they utilized two techniques, namely TF-IDF , to prepare the text-based features and several ML classiﬁers, such as Decision Tree, Naive Bayes, AdaBoost, Logistic Regression, K-nearest neighbor, Support Vector Machines, and Random Forest, to predict whether an email is phishing. Two imbalanced datasets (4082 non phising & 501 phishing emails and 5088 non phising & 612 phishing emails) were deployed to measure the eﬀectiveness of the classiﬁers, using the accuracy as a metric, and the results indicate that the ML classiﬁers performed better.

The drawback of this work is that the authors did not use a metric that is suitable for imbalanced data, such as F1-score. Instead, they measured the eﬃcacy of their approach using accuracy, which is biased towards the majority class (namely, the class that contains the most samples, which in their case is the non phising emails).

The work in this paper improves the approach that presented in [40] (a) by performing feature selection before the classiﬁcation process to identify the features that contribute to better classiﬁcation performance, (b) by deploying Word2Vec and BERT techniques, which are new and more well-known in text classiﬁcation tasks than Doc2Vec, (c) by utilizing F1-score metric that depicts the model’s performance when tested in imbalanced data more precisely, and d) by considering the evolution of phishing emails using only new phishing emails. In [13], the authors presented a phishing detection framework that is based on Recurrent Convolutional NNs, named THEMIS.

The Word2Vec method was utilized to obtain the vector sequences from the character-level and word-level of both the emails’ header and body ﬁelds. THEMIS accomplished 99.848% detection accuracy with a 0.043% FPR. Towards the same direction, the method proposed in [27] combined Convolutional NNs and Keras Word Embedding to detect phishing emails focusing on the text. The authors compared two datasets, one with email headers and one without.

The results showed that the model achieves higher detection accuracy (96.8%) when the email headers are not taken into account. A drawback of both [13] and [27] is that the authors did not consider the evolution of phishing emails, which is a signiﬁcant limitation as the phishing emails have evolved over the years. Furthermore, another limitation of [27] is that the authors deployed an imbalanced dataset (4082 non phising and 501 phishing emails) in their experiments, and measure the classiﬁers’ performance only on the classiﬁcation accuracy without utilizing other metrics (e.g., F1-score, AUC) that depict better the performance on imbalanced data.

2.2 Proposed System

Several Machine learning-based approaches can be utilized to enhance the efficiency of anti phishing systems. This is achieved by implementing an enhanced predictive model emphasising the

optimal selection of feature vectors extracted from online elements like URLs, webpage properties, and webpage behaviour. The methodology involves an incremental component-based system that presents the feature vectors to the predictive models, utilizing both Support Vector Machine (SVM) and Naïve Bayes (NB) algorithms. These algorithms were tested on phishing and non phising datasets, achieving an impressive accuracy of 99.96%.

a: DECISION TREE (DT)

A commonly used ML algorithm that can be applied for regression and classiﬁcation is the decision tree. A recursive partitioning algorithm is applied to test the availability of attributes or features considering speciﬁc purity indexes. The Gini Index and Entropy are the most commonly used indexes, where the former is applied to measure the probability of a randomly chosen feature that is incorrectly classiﬁed [49]. The uncertainty amount that is proportional to the information gain is referred to as Entropy [49]. By means of these indexes, the required position of the features, either internal node or root, can be determined.

b: SUPPORT VECTOR MACHINES (SVM)

SVM is usually applied for classiﬁcation activities as well as regression activities. Each data item within the SVM is plotted as the point within the n dimensional space (n is the feature number for each sample within the training set). The mission of the algorithm is to extract the most appropriate hyper-plane which can be split into two classes. The non linearly separable data is classiﬁed by SVM through transformation into higher dimensional space, with the help of a kernel function, in which a separating hyperspace is present. Yet, it is difﬁcult to interpret the SVM, and it is quite memory sensitive.

c: NAÏVE BAYES (NB)

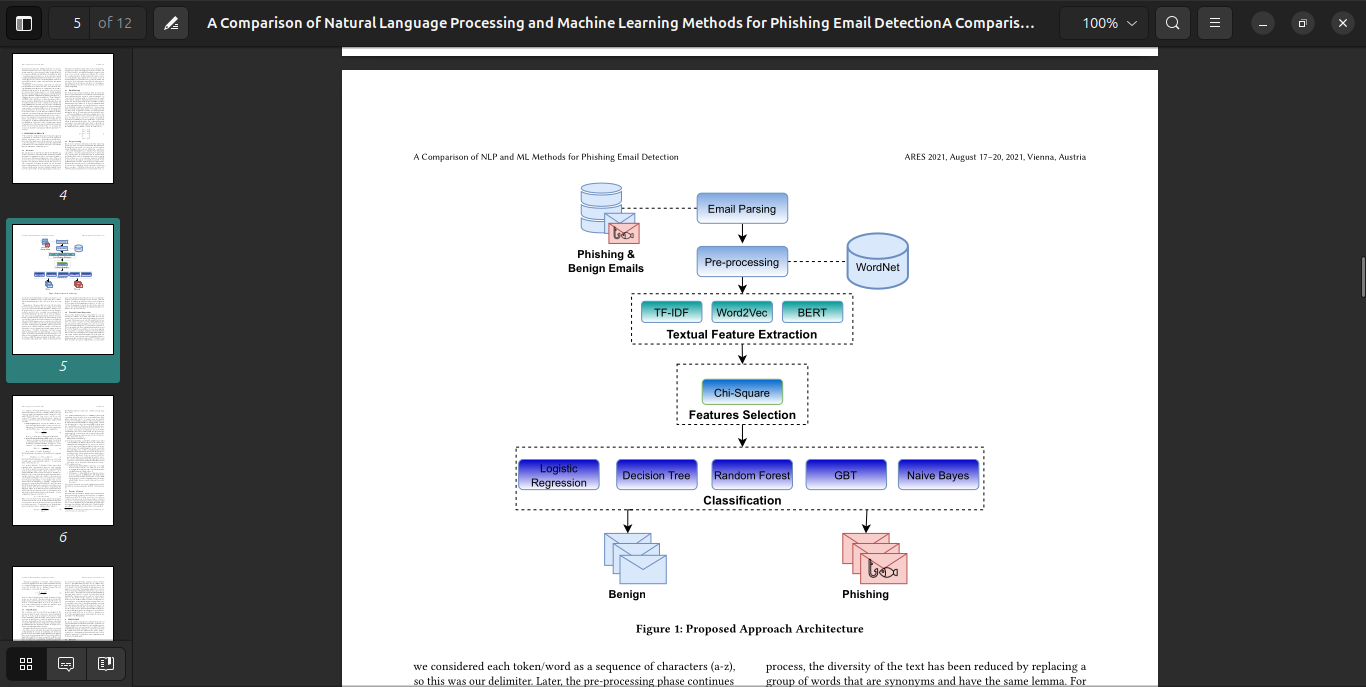
The Bayes rule of conditional probability is applied by this classiﬁer, and all data features are applied. They are individually analysed based on the assumption that they are not only independent but also as important as one another. Quick convergence and simplicity are the classiﬁers beneﬁts, yet it is not possible to understand the associations and interactions amongst the features of each of the samples.

g: K-NEAREST NEIGHBOURS (KNN)

An applied supervised learning algorithm is the KNN, which usually helps in classiﬁcation. The assumption here is that similar aspects maintain close proximity.Similarity measures are applied to check for the similarity degree, most commonly the Euclidean distance.

Implementation is easy with KNN, as tune parameters and model parameters are not built. The KNN is referred to as a non-parametric algorithm, which is why fundamental assumptions regarding the distribution of data are not required. The algorithm will perform slower based on the increase in size and dimensionality of the dataset.

In the Figure 1, the architecture of the proposed approach is presented. The email parsing task associates the extracted body text of emails with their respective class and places them in a matrix. The pre-processing task is responsible for cleaning the emails’ text and converting them to a uniform format. To this end, the texts are converted into lowercase, and the special characters, stopwords, and punctuation marks are removed.we considered each token/word as a sequence of characters (a-z), so this was our delimiter. Later, the pre-processing phase continues with the lemmatization phase that is the core of the pre-processing task.

Figure 1: proposed system architecture

**CHAPTER 3**

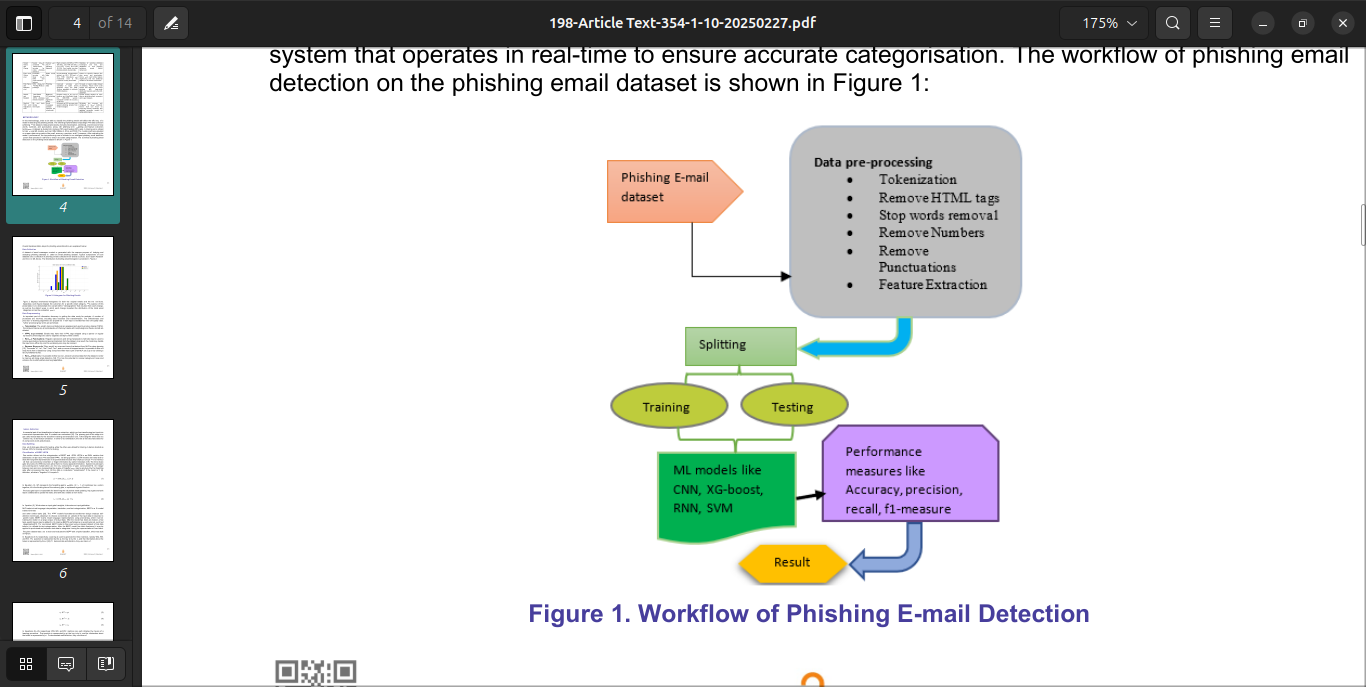
**METHODOLOGY**

In this methodology, order to be able to classify the phishing emails will affect the efficiency of a

model in detecting the phishing emails. The following implementation steps begin with data collection (phishing Email dataset):

* **Data Preparation:** We use publicly available email corpora containing both legitimate and phishing messages. Emails are pre-processed (tokenization, normalization, etc.) to prepare text data for analysis.
* **Feature Extraction:** For classical ML, we extract features (e.g. TF-IDF word vectors, sender/domain indicators) from the email subject and body.
* **Model Training:** We train several classical classifiers (e.g. support vector machines, random forests, gradient boosting) and decision trees on the labeled training data.
* **Evaluation:** Each model is evaluated on a held-out test set using accuracy, precision, recall, and F1-score to measure detection performance and false positive rates.
* **Deployment:** The best-performing model is integrated into a Streamlit web application, providing an interactive demonstration interface for real-time phishing email classification.

The workflow of phishing email detection on the phishing email dataset is shown in Figure 2:

Figure 2: Workflow of Phishing E-mail Detection

**3.2 Data Preprocessing**

An important part of information discovery is getting the data ready for analysis. A number of

processes are involved, including data reduction and transformation. The effectiveness and

precision of learning algorithms are jeopardised if raw input is transformed into low-quality data.

Further processing key terms are as follows:

• Tokenization: The email's topic and body text are analysed and used to produce tokens [13][14].

The retrieved tokens are all normalised such that any tokens with morphological or flexional ends are deleted.

• HTML tags removal: Emails may have their HTML tags stripped using a parser or regular expressions; these tags are used to organise and style online content.

• Remove Punctuations: Regular expressions and string manipulation methods may be used to

remove punctuation marks and special characters from the dataset. As a result, the model may handle the input more efficiently, and the vocabulary size may be reduced.

• Remove Stop words: "Stop words" are overused terms that detract from NLP for deep learning

[15]. The words "a", "an", "the", "and", "but", and so on are all stopped words. It is possible to filter out stop words from a dataset by using a stop word filter that is part of an NLP package or by utilising a list of prohibited words.

• Remove Numbers: It is possible to filter out non-relevant numerical data from the dataset in order to improve phishing email detection [14]. This has the potential to reduce background noise and enhance the model's pattern-learning capabilities.

**3.3 Feature Extraction**

An essential part of text classification is feature extraction, which involves transforming text input into a numerical representation that DL models can understand [16]. The primary goal of this stage is to get useful textual data for the classifier's training and evaluation [17]. Tokenising the whole text is a common way to do feature extraction. In order to do tokenisation, the text is first deconstructed into its component words and phrases.

**3.4 Data Splitting**

One set of data was utilised for testing, while the other was utilised for training. A data is divided as follows: 80% for training and 20% for testing.

**3.5 Programming Language**

The programming language selected for implementing the proposed phising detection system is

Python. Python is widely regarded as one of the most suitable languages for machine learning and

deep learning applications due to its simplicity, readability, and extensive ecosystem of libraries

and frameworks.