National College of Ireland

Deep Learning Phishing Email Classifier Combined with NLP

MSc Research Project
Cybersecurity

EBONG, MAURICE ANIEFIOK
Student ID: 20148127

School of Computing
National College of Ireland

Supervisor: Dr. Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: … EBONG, MAURICE ANIEFIOK……

Student ID: …20148127……

Programme: …Master of Science, Cybersecurity…… **Year:** …2022……..

Module: …Research Project……

Supervisor: …Dr…Vikas…Sahni……

Submission Due Date: …26TH APRIL, 2022……

Project Title: … A Deep Learning Phishing Email Classifier Combined with NLP…

Word Count: …7822…….. **Page Count**…23……

I hereby certify that the information contained in this (my submission) is information pertaining to the research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other authors' written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……

Date: ……26th April 2022……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	☑
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☑
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on the computer.	☑

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Deep Learning Phishing Email Classifier Combined with NLP

EBONG, MAURICE A.
20148127
M.Sc. in Cybersecurity
8TH APRIL, 2022

Abstract

Given the fact that phishing email attacking techniques are constantly being developed and updated, the current methods are inadequate to tackle the issue. Additionally, the increase in the number of attacks mostly suggests that there is a need for the development of robust techniques in tackling phishing email attacks. One primary concern with phishing email detection is that the current phishing detection technique cannot adapt to the ever-changing methods and semantics used by phishers (attackers) against their victims. In this research work, two machine learning techniques namely Support Vector Machine (SVM) and Random Forest (RFC) and a Deep learning technique namely Deep Neural Network (DNN) had been used to classify phishing emails. NLP word2vec technique was applied to the dataset and resampling techniques were also applied to the dataset to handle the imbalance in the dataset. The results obtained from the models implemented indicate that SVM, RFC, and DNN have 100% accuracy in classifying phishing emails and recorded a training and testing time for models are 133.3s and 0.21s, 943.70s and 0.09s, 436.85s and 2.29s respectively.

1 Introduction

The use of technology and social engineering to pilfer data related to individuals' identities and accounts has seen a significant increase, especially with association with phishing attacks (Salloum et al., 2021). Phishing emails are usually used to send links to websites or attachments where users are prompted to provide sensitive personal information such as banking, financial, or login credentials (Rawal et al., 2017). Thus, phishing is a primary element of many cyber-attacks and is usually the first step in advanced persistent threats.

1.1 Research Motivation and Background

Phishers use various methods to initiate attacks, the most popular methods are email, SMS, and social media, and are perpetrated by sending emails to large numbers of individuals. Because of the mass adoption of emails as primary means of communication, especially by corporations and

businesses, there is an endless pool of potential victims (Burita et al., 2021). Also, phishing attacks are largely successful because of the volumes of messages sent. In 2006, a group of hackers based in the US used emails to deceive individuals into providing login credentials to online accounts. Over a decade later phishing techniques have advanced, aggravating the difficulty in identifying fraudulent emails. Verizon in 2016 data breach reported about 636,000 phishing emails sent, of which only 3% of the targets reported to the management of the possible fraudulent activity.

Additionally, phishing emails have become more difficult to identify because of the targeted approach used by phishers called spear phishing. Spear phishing is an attack that is targeted toward a specific individual, organization, or business. Another problem is that tracking criminals are difficult because of the masking of digital identities. The methods and techniques used by attackers or phishers are constantly evolving and so are the detection techniques presented to keep end-users safe (Rastenis et al., 2021).

Email systems are susceptible to spoofing and phishing attacks because of the availability of valuable email messages. To truncate the threats of phishing emails, a variety of solutions have been proffered. Many research exploring the use of behavioral patterns exist for detecting and preventing phishing attacks like machine learning. Machine learning is one of the most commonly used methods for phishing detection and prevention. As recurring attributes are identified, automated tools have also been developed to aid users and cyber-security enterprises to identify malicious emails (Rastenis et al., 2021). Behaviors such as changes in the interface, color, domain, and redirect verification are identified through machine learning (Senturk et al., 2017).

1.2 Research Question

Phishing email detection is said to be efficient if the classification is accurate, execution time is shortened, and errors and false classifications are kept low in addition to low storage requirements (Salloum et al., 2021). The NLP detection mechanism is based on the examination of changes through semantics. Therefore, I aim to answer the following question.

RQ*: "What impact does NLP using word embedding techniques on machine learning techniques (Support Vector Machine (SVM) and Random Forest Classifier (RFC)) and deep learning technique (Deep Neural Network (DNN)) have on phishing email detection models and what are the difference in time?"*

The impact of the models will be measured using evaluation metrics such as accuracy, AUC score, and precision.

1.3 Research Objectives

To adequately respond to the research question of this research work, this report aims at actualizing the following set objectives:

- Critically review existing research work on phishing email classification using different machine learning and deep learning techniques other researchers have implemented. This will be important and it provides useful information on the approach to be adopted in the cause of carrying out this research work.
- From the critical review, adopt a known data mining methodology that will guide the approach used in carrying out this research work.
- Pre-process the datasets used in the implementation of this research work which might in but is not limited to handling missing data, handling raw data, etc., and splitting the dataset into train dataset and test dataset.
- Develop, implement and train models (both machine learning and deep learning models) on the training dataset and evaluate the performance of the trained models on the test dataset.
- Compare the performance of the developed models with existing models, and also compare their processing time.

The remaining part of the research report is organized as Chapter 2 discusses Literature Reviewed; Chapter 3 describes the methodology adopted in carrying out the research work. Chapter 4 covers the implementation, result, and analysis obtained from the implemented model, chapter 5 describes the discussion of the implementation, and chapter 6 discusses the conclusion of the research project.

2 Literature Review

2.1 Phishing Emails Detection and Classification Approaches

Salloum, Gaber, Vadera, and Shaalan (2021) surveyed phishing occurrences in cyberspace. The work featured dimensions of phishing, emerging solutions to phishing emails, and a comparative perspective to existing phishing solutions. Salloum et al. (2021) focused on email communication identified as the most common phishing means. Preventing or controlling phishing requires that the attack is identifiable. Thus, it is arduous work to identify a phishing attack. Salloum et al. put forward blacklisting, machine learning-based algorithms, and deep learning as mechanisms to identify phishing attacks. The deep learning approach is best favored in the survey because it yields phishing identification without sophisticated cyber security expertise.

Natural Language Pro0cessing alongside machine language has developed effective strategies to combat phishing. However, Salloum et al. indicated that NLP's major flaw in countering phishing is its major reliance on texts from email surfaces rather than deep semantics. Salloum et al. emphasized that relevant studies have been conducted on combating phishing but qualities of easy interpretation and deep-level insight into legitimate and phishing emails are still lacking. Existing solutions to phishing are inadequate. Thus, comprehensive solutions are required to combat the rising phishing debacles in cyberspace (Salloum et al, 2021). White (2021) found that the deep learning model for identifying phishing was most accurate among ML models, SVM, Naïve Bayes, Long Short Term Memory (LSTM), and DL. ML performed better in computation time while CNN performed better in accuracy using semantic analysis for phishing detection.

Verma and Hossain (2014) considered NLP's usefulness in identifying phishing emails from a text-based perspective. The work focused on using a general semantic feature for solving phishing email problems. Verma and Hossain developed a semantic and statistical-based body text classifier that generated 95 percent accuracy in identifying phishing emails; reduces vulnerability and covers for frequent retraining required for classifiers in machine learning. The results in the study are recommended for cross-validations for future implications. Yasin and Abuhasan (2016) conducted a study on similar work to Verma and Hossain (2014).

The study focused on the Model of Knowledge Discovery which involves data mining essential for building an intelligent model that distinguishes emails as legitimate or spam. Five classifiers were applied for analyzing the data set of emails to detect phishing and cross-compare the accuracy of each software. The Random Forest Algorithm and J48 generated the highest accuracy for the analyzed dataset with 99.0% and 98.4% accuracy respectively. Rawal, Rawal, Shaheen, and Malik (2017) realized a higher result for accuracy in phishing detection using the Random Forest Algorithm applied by Yasin and Abuhasan (2016). Rawal et al. generated higher accuracy for phishing detection using machine learning techniques involving SVM and the Forest Random Algorithm. Burita, Matoulek, Halouzka, and Kozak (2021) adopted different software to analyze 200 emails for phishing. The use of Tevek for phishing detection in the study contributed to an alternative technique for identifying phishing in differing forms and having knowledge on preventing phishing attacks.

2.2 Phishing Detection and Spam Emails Identification

Rastenis, Ramanauskaite, Suzdalev, Tunaityte, Janulevicius, and Cenys (2021) analyzed 700 phishing and spam emails toward automated security in cyberspace. The work covered a multi-language perspective that other works have not covered. Russian and Lithuanian were used in dataset gathering and for the analysis process. The study generated 89.2% accuracy in detecting phishing and spam emails. Importantly, when the 700 datasets included in the study were translated to Russian and Lithuanian, the same accuracy rate was recorded. However, the accuracy dropped to 77.0% when SpamAssassin and Nazario datasets were included in the analysis. The accuracy decrease indicated that time differences, region, and organizational profile of the used data influenced the generated accuracy.

Deep learning solutions are suggested for further studies on automated security in email communication. House (2013) considered the influence of user responses to phishing attacks. The study showed that participants who had a high fear arousal level were less likely to intend or respond to phishing emails. The psychological consideration in the work also revealed the significant impact of self-confidence in avoiding vulnerability to phishing links. Although, participants with high self-confidence were not shown to be more vulnerable to phishing attacks those with a high fear level were less vulnerable due to reduced intention to reply to phished emails.

On the techniques to verify emails for phishing, Shekokar, Shah, Mahajan, and Rachh (2015) proposed a novel form of verifying phishing. Shekokar et al. used URL-based and webpage

verifications to determine phishing content. LinkGuard Algorithms are adopted for link analysis. The actual link is extracted from the phished link to identify the possibilities of fraud in email communication. Shekokar et al. provided that the inadequacy of LinkGuard Algorithms to identify phishing from link verification is complemented in visual comparison of web pages. The algorithm determines phishing through visual similarity if the link comparison becomes insufficient. Senturk and Yerli (2017) combined machine learning and data mining techniques for phishing detection and protection. The integration of ML and data mining resulted in 89% phishing detection capacity in the study. Further phishing detection techniques are suggested including Machine Learning Anti Phishing Technique (MLAPT) which observed emails behaviors to detect phishing, Black Listing Mechanisms (BLM) such as Simple Mail Transfer Protocol (SMTP), and pattern recognition by extracting phishing features for classification and identification. Senturk and Yerli identified that existing phishing detection and protection techniques proposed in phishing literature are processes-centered and maybe capital intensive in work areas with high email exchanges.

2.3 Machine Learning Approach to Phishing Email Detection

Using a machine learning approach, Basnet, Mukkamala, and Sung (2008) conducted a study on phishing attacks detection. The study is one of the studies that focused on algorithms used in machine learning to classify spurious and legitimate emails. Key features of the phished messages are classified by machine learning algorithms. The study adopted 4000 emails in two datasets and classified them as ham and spoofed emails. ML learning methods including Support Vector Machines (SVM) (Biased SVM, Leave One out Model), Neural Networks, Self Organizing Maps (SOMs), and K-means were used for the study. SVM provided the highest accuracy among the ML methods while BSVM and Artificial Neural Network returned the same accuracy output of 97.99%. In the same vein, Kumar, Chatterjee, and Diaz (2019) used SVM combing feature extraction and classification for phishing detection. Kumar et al. adopted a similar binary approach as used by Akinyelu and Adewumi (2014). The hybrid method in the study yielded 98% accuracy for phishing recognition. Ozcan, Catal, Donmez, and Senturk (2021) also implemented a hybrid method involving a deep learning model. NLP and character embedding were combined to detect phishing. The study focused on deep learning because of its extreme ability to learn from the dataset and categorize the phished email communication challenges. The hybrid model experimented in the study performed better than existing detection models.

Fang, Zhang, Huang, Liu, and Yang (2019) developed a novel approach to detecting phishing emails. The approach, THEMIS, combined multilevel vectors, attention mechanisms, and Recurrent Convolutional Neural Networks (RCNN). THEMIS generated 99.848% accuracy in detecting phishing emails and identifying harmless messages. Using a dataset from the First Security and Privacy Analytics Anti-Phishing Shared Task, THEMIS was found effective in recognizing phished email messages. Zhang, Chen, and Huang (2018) also experimented with RCNN on attention-centric classification for phishing identification. RNN showed the capacity for learning context and temporal features while CNN showed the capacity for catching potential features. Zhang et al. used an attention mechanism to ensure that the model paid attention to the

information in the experimentation processes. The SemEval-2010 Task8 public dataset was analyzed in the study with TensorFlow, a deep learning tool. Although RCNN showed improvement in classification, it only performed better when no semantic feature is included.

2.4 Algorithm-based Phished Email Classifier

Akinyelu and Adewumi (2014) classified emails as phishing and legitimate through an algorithm-focused approach. The study included the use of Random Forest (RF) as a machine-learning algorithm to classify emails. 99.7% phishing detection accuracy was realized. Akinyelu and Adewumi concluded that future phishing will focus more on semantic attacks than syntactic attacks; thus, security automation in cyberspace should be increasingly charged with semantic-focused social engineering protection. A 97.14% accuracy result was generated when Mahajan and Siddavatam (2018) used the same ML approach on a selected dataset. The classifiers performed better when more data were allotted to training and the integration of RF in ML and blacklist methods are recommended for further studies in phishing detection. Sonowal (2020) using a binary approach to studying phishing email detection found that BFSF required the least time in feature evaluation for phishing detection when compared to other methods. The study centered on the email subject, the body of the email, hyperlinks, and content readability with an accuracy result of 97.41% generated through BFSF. Gutierrez, Abri, Armstrong, Namin, and Jones (2020) used embedded emails for classification in detecting phishing emails. The study showed that email embedding resulted in more effective classifiers to identify phished emails. Gutierrez et al. used Doc2Vec for embedding documents and RF, SVM, Logistic Regression, and Naïve Bayes for classifying emails as legitimate or dubious. SVM returned the highest accuracy rate.

Siddique, Khan, Din, Almogren, Mohiuddin, and Nazir (2021) generated a 98.4% accuracy rate using CNN, SVM, LSTM, and Naïve Bayes based on ML algorithms. LSTM outperformed other methods in the experiment. Daniel, Reshma, and Selvarani (2021) introduced a new approach to detecting phishing URLs. The study identified lexical features of links as significant in distinguishing phished links from genuine links. The URL length, brand names, and brand popularity are lexical features considered for identifying phished content. Zalavadia, Pandey, Pachpande, Nevrekar, and Govilkar (2020) adopted deep learning frameworks including neural networks for semantic analysis of texts in detecting phishing emails. Zalavadia et al. combined ML algorithm techniques for the experimentation and discovered that Random Forest (RF) and Extra Trees (ET) performed better. NLP contributed to semantically noticing suspicious and inappropriately structured texts to recognize phishing. Similar to the approach used by Zalavadia et al., Lansley, Kapetanakis, and Polatidis (2020) implemented a semi-synthetic dataset to identify malicious content in social networks. The NLP-based study identified possible phished messages by ML algorithms. Intent, spelling, and links were analyzed to determine if phishing attacks were possible or not. Lansley et al. identified that NLP alongside ML proved more effective than alternative strategies in detecting phishing contents. Deshpande, Pedamkar, Chaudhary, and Borde (2021) added that address bar features, abnormal-based features, HTML and JavaScript features, and Domain-based features are functional for detecting phishing emails in NLP and ML methods.

Sujithra, Dwivedi, and Utakarsha (2020) also adopted ML algorithms for phishing detection. XG Boost outperformed Neural Networks and Logic Regression (LR) in classifying social messages as real or harmful.

2.5 Summary of the Reviewed Literature

The reviewed literature shows that many studies have been conducted on phishing detection in social networks using a machine learning model. Few works were found in the existing literature on phishing with a focus on deep learning for detecting phished messages. Machine Learning (ML) and Natural Language Processing (NLP) were the most combined approach to phishing classification in the existing literature. Rastenis et al. (2021) expanded the scope of phishing detection by translating their dataset to Russian and Lithuanian to give more international-centered security in the social engineering space.

Most reviewed studies in the literature focused on semantic analysis through NLP for classifying email messages as genuine or spurious with the exception of Daniel et al. (2021) where the study was on using lexical features for detecting phishing contents. Few works were found on NLP and Deep Learning (DL) combinations for recognizing malicious messages. Thus, this study focuses on deep learning phishing email classifier combined with NLP to fill the existing knowledge gap in phishing literature. This study is significant because DL has been shown in the literature to give improved data behavior understanding and fewer technicalities in information classification. The findings in this study will enable even non-technical email and other social network users to detect phishing emails and avoid fraudulent attacks.

Author(s) and Publication	Methodology Employed	Dataset	Advantage	Limitation	Performance Analysis
Bagui, S., Nandi, D., Bagui, S., and White, R., J. (2021)	Naïve Bayes, SVM, Decision Tree, as well as DL models, Convolutional Neural Networks (CNN) and Long Short Term Memory (LSTM)	18,366 labeled emails, of which 3,416 were phishing emails and 14,950 were regular emails	Implements deep semantic analysis for identifying text features in classification	Generally focuses on deep semantic analysis	Deep learning beats machine learning in classification analysis while machine learning leads in computation time experiment
Burita, L., Matoulek, P., Halouzka, K., and Kozak, P. (2021)	Text analytical software Tevek	200 email messages	Improving knowledge, understanding, and training on email defense	Less dataset involved in the analysis	The developed model comparatively proves more effective than existing models.

Daniel, A. J., Reshma, G., and Selvarani, C. (2021)	Random forest and Support vector machine algorithms	The traffic rank feature is acquired from Alexa.com	The implemented model reduces latency and strengthens security in email communication	The analysis is only from the lexical perspective.	Time improvement and response accuracy were found in the model.
Deshpande, A., Pedamkar, O., Chaudhary, N., and Borde, S. (2021)	HTML, CSS, JavaScript and Django	Unstructured data of URLs from Phishtank website, Kaggle website, and Alexa website	Enabled automated detection of phishing contents.	URL manipulation may escape automated detection of phishing links.	Regular URLs are detectable and distinguishable for phished messages.
Fang, Y., Zhang, C., Huang, C., Liu, L., and Yang, Y. (2019)	Convolutional neural networks (RCNN) model with multilevel vectors	Enron dataset and SpamAssassin	Proposed a new phishing detection model, THEMIS	Inclusion of unbalanced dataset in phishing analysis	THEMIS responds positively to phishing detection using the email header, body and text
Gutierrez, L. F., Abri, F., Armstrong, M., Namin, A., and Jones, K. S. (2020)	SVM, Logistic Regression, and Random Forest	Twenty-four email stimuli were created for an experiment with human subjects.	Proposed email embedding technique as an alternative solution to phishing	Data was subject to human control thus affecting accuracy and objectivity.	SVM model accurately classifies the data as malicious of genuine.
Kumar, A., Chatterjee, J. M., and Diaz, V. G. (2019)	Hybrid Methodology	1705 emails out of that 1291 area unit ham and 404 area unit phished.	Advanced feature extraction from texts and images for classification	Less dataset was applied.	Accuracy found in feature extraction and classification using Tree Model and SVM.
Lansley, M., Kapetanakis, S., Polatidis, N. (2020)	ML and NLP	semi-synthetic dataset	Checks intent, links and spellings for spurious contents.	The approach's recognition pattern is text-based.	Textual recognition of links was found effective.
Mahajan, R., and Siddavatam, I. (2018)	Decision Tree, random forest and Support vector machine algorithms	The data set consists of total 36,711 URLs which include 17058 benign URLs and 19653 phishing URLs.	Applied model found effective in classifying data for improved phishing recognition	Analysis focused on classifier's performance.	ML method used performed better as classifiers.
Ozcan, A., Catal, C., Donmez, E., and Senturk, B. (2021)	Hybrid deep learning The model includes machine learning methods	Ebbu2017 a secondary Dataset built from several Internet resources.	Consolidation of character-embedding manual and Natural Language Processing (NLP) in feature extraction	Focused on extraction features for classification in phishing detection.	The hybrid learning model proves more effective than existing models.

	(i.e., k-Nearest Neighbors (kNN) and tree-based methods) and deep learning algorithms (i.e., RNN and CNN-based methods)				
Rastenis, J., Ramanauskaite, S., Suzdalev, I., Tunaityte, K., Janulevicius, J., and Cenys, A. (2021)	Naïve Bayes, generalized linear model, fast large margin, decision tree, random forest, gradient boosted trees, and support vector Machines methods	700 spam and 700 phishing emails	Automated classification of spam and phishing emails.	Deep neural networks not involved in the analysis	Only 3 out of 7 analyzed solutions solved the accuracy problem in phishing detection
Salloum, S., Gaber, T., Vadera, S. and Shaalan, K. (2021)	Survey Analysis of Natural Language Processing (NLP) and Machine Language (ML)	Survey analysis of phishing rejection research papers	The study shows the relationship between phishing email detection and NLP techniques.	The survey focused only on the NLP technique and ML strategies for phishing detection	The amalgamation of surveyed phishing detection using NLP techniques.
Siddique, Z, B., Khan, M, A., Din, I, U., Almogren, A., Mohiuddin, I., and Nazir, S. (2021)	Naive Bayes, CNN, SVM, and LSTM	Raw data collected is obtained from the online resource Kaggle.	Analyzed email contents in English, Russian, and Lithuania.	Analyzed data from a textual perspective.	An equal accuracy rate was generated across the experimented languages.
Sonowal, G. (2020)	Binary search feature selection (BSFS) with a Pearson correlation coefficient algorithm	2500 phishing and non-phishing emails	Improves time requirement for accuracy response in phishing detection.	Focused only on the binary application for experiment and analysis.	Binary search feature selection outperforms other search feature models in the experiment.
Sujithra, T., Dwivedi, N.,	Machine Learning, Deep Learning, and	ML algorithm for website testing	ML algorithm is used to reduce false positives in phishing site detection.	Limited to Web sites in	XG Boost outperformed other

Utakarsha, A. (2020)	Support Vector Machine (SVM)			experimentation.	classifiers considered.
Zaladavia, F., Pandey, S., Pachpande, P., Nevrekar, A., Govilkar, S. (2020)	Deep Learning frameworks with Neural Network	SpamAssassin and Ham-Spam datasets	Uses semantic analysis for intelligent recognition of malicious contents.	The NLP approach is limited to the DL framework.	Random Forest and Extra Trees outperform SpamAssassin in the conducted experiment.
Zhang, X., Chen, F., and Huang, R. (2018)	RNN and CNN (RCNN)	SemEval-2010 Task 8 Dataset	Temporal and long-term feature learning is implemented.	The only neural network is considered for the study.	The combined neural networks proved more effective than the existing models.

Figure 1: Comparative Review of Related Work

3 Research Methodology

CRISP-DM (Cross Industry Standard Process for Data Mining) methodology for data mining was modified and adapted in order to develop a structural approach in the implementation of this research effort, which is in line with the specified objectives. The CRISP-DM methodology defines a six-step iterative framework for any data mining project, although only five of the iterative stages will be used in this research work, excluding the deployment stage. The CRISP-DM framework is illustrated in Figure 1 below, as it was employed in the research work case study.

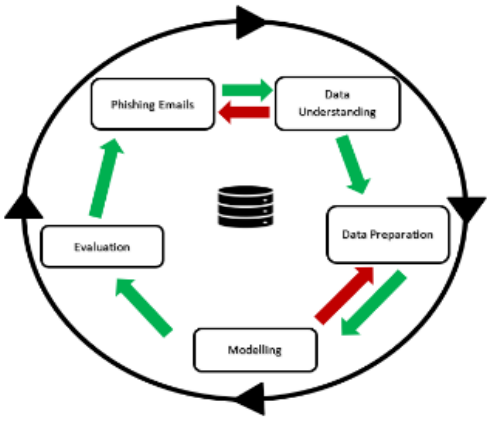

Figure 2: Modified CRISP-DM iterative process model.

3.1 Phishing Emails

This section focuses on the research challenge of phishing email classification as well as the stated research objectives. Previous efforts to address the issues of classifying an email as phishing or

legitimate email led to the development of the research objectives. This research focuses on addressing challenges associated with phishing email classification by employing well-known machine learning and data mining techniques, with the goal of developing an optimal solution with increased efficiency in identifying phishing emails.

3.2 Data Understanding

To achieve the set objectives for this research work, Nazario's phishing email corpus downloaded from ***monkey.org***, and Enron's email dataset downloaded from the ***Carnegie Mellon University School of Computer Science*** website were considered to be significantly relevant to the subject being researched. The Nazario phishing email corpus contains over 1300 phishing email records. Enron's email dataset contained over 500,000 legitimate emails records.

3.3 Data Preparation

Feature extraction and Feature selection will be performed on the dataset. In order to carry out feature extraction and feature selection, Python Natural Language Toolkit (NLTK) library will be used to parse and tokenize the formatted corpora, remove stopwords, and perform stemming operations. After feature extraction and feature selection, the dataset will be divided into a train set and a test set in the ratio of 7:3 (i.e., 70% of the data will be used to train the various model to be implemented and the remaining 30% will be used for testing the developed model) respectively. To manage the imbalance in the dataset SMOTE was applied to the split dataset to eliminate bias in the developed model

3.4 Modeling

Deep Neural Network (DNN) is the deeply learned technique deployed to classify emails as phishing emails or legitimate emails. Support Vector Machine (SVM) and Random Forest (RF) classifier were the machine learning techniques implemented as these techniques are best for classification problems.

3.5 Evaluation

In this phase, a comparative analysis was carried out to measure the performance of the techniques implemented for this research by using Accuracy, Precision, and Area under the ROC Curve (AUC) which is a summary of how well the techniques were able to classify phishing correctly.

4 Implementation, Evaluation, and Analysis of Results of Email Phishing

The goal of this section is to describe the various processes and activities carried out in the classification of phishing emails to achieve the set objectives of the research paper. The project implementation and evaluation code are done with Python programming language version 3.6 using Jupyter Notebook IDE from Anaconda, a Python programming language package manager. Also, 3 metrics different will be used to measure the performance of the proposed models to be deployed in this project, and these include

- Accuracy – is the measure of the ratio of correctly classified phishing emails and legitimate emails.
- Precision – is the measure of the ratio of True Positive (TP) to the total sum count of True Positive (TP), and False Positive.
- (Area Under the Curve ROC (AUC) – is a measure of the ability of the model to distinguish between classes. Dataset and Data Pre-processing

More details are included in the configuration manual.

4.1 Exploratory Data Analysis of Phishing Emails

Python's Matplotlib library was used to visualize the relationship between the dependent variable to determine that the binary class property (emails are either phishing email or legitimate) of the emails is evenly distributed. The Pie plot labeled Figure 3 below shows that 87.5% of emails belong to the legitimate email class and 12.5% of emails belong to phishing emails

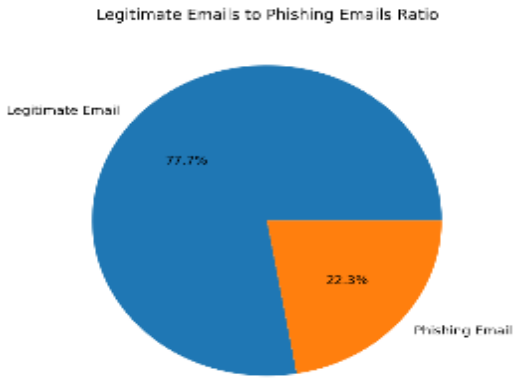

Figure 3: A Pie chart showing the frequency distribution of legitimate emails to phishing emails in the dataset

Python's WordCloud library was also used to plot/show the most common word in the two classes of emails. Figure 4 below shows the word cloud plot for legitimate emails while Figure 5 below shows the word cloud plot for phishing emails.

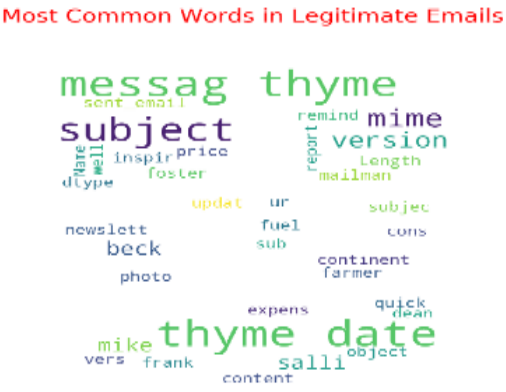

Figure 4: A word cloud plot showing the most common tokens in the class of legitimate email

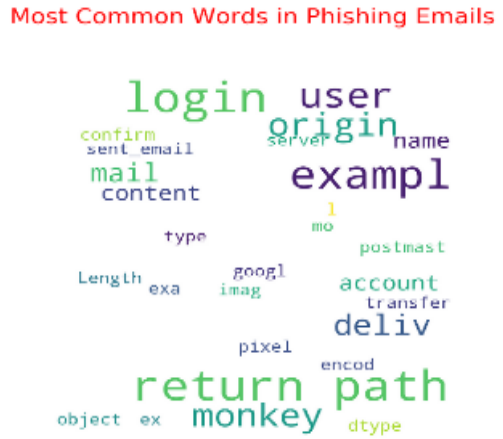

Figure 5: A word cloud plot showing the most common tokens in the class of phishing email

Figure 5, shows some of the keywords in the phishing email class to be login, banks, account, etc. The term phishing is synonymous with user data theft and the keywords shown to agree with the common words. Figure 4 shows the legitimate common words such as message, subject, bill, date, bank, etc.

4.2 Feature Engineering and Modelling

In the visualization of the data records in the ***phishing_email_df*** dataframe, the dependent variable shows a high degree of imbalance in the dataset as 22.3% of a data record are grouped as phishing email while 77.7% is grouped as legitimate emails. This imbalance will result in bias classification of phishing emails as legitimate emails if the imbalance in the dataset is not removed. To the issue of class imbalance in the dataset, SMOTE techniques were applied to the dataset to use random oversampling techniques to create equal or even class distribution between the phishing email class and the legitimate email class.

Two categories of experiments were performed on the dataset, these include the use of supervised learning techniques such as SVM and RandomForest Classifier and the use of Deep Neural Network (DNN) as a deep learning approach in an effort to be able to classify the email as phishing emails and legitimate emails. Helper functions and Custom types were created to ensure the coding was concise and reusable for easy comprehension and to encourage code reuse. The models being used in the classification of works with numeric data, hence Pandas' ***get_dummies()*** method was used to convert the dependent variable to 0 or 1 where 0 represented the class of legitimate email and 1 represented the class of phishing email. The words in the sent email dataframe were converted to numerical encoding using Python's CountVectorizer from the sklearn feature extraction library. The use of these different libraries alongside the helper functions was used to lead to the evaluation and presentation of the outcome of each model in the analysis of the dataset.

4.3 Implementation, Model Analysis, and Result

Three different models were implemented using two machine learning algorithms namely the Support Vector Machine (SVM) algorithm and Random Forest and the third model is a Deep Neural Network (DNN) is a deep learning algorithm.

4.3.1 Support Vector Machine (SVM) Implementation, Analysis, and Result

Support Vector Machine (SVM) is a well-known Supervised Learning algorithm, which is used for Classification as well as Regression problems. SVM algorithm is primarily used for Classification problems in Machine Learning.

4.3.1.1 Implementation

Two sets of experiments were carried out on the prepared dataset. In carrying out these experiments, the SVM module was imported from the sklearn Python library. More details are added to the configuration manual.

4.3.1.2 Result and Analysis

	SVM	TUNED SVM
Accuracy	99.83	100.0
AUC	99.83	100.0
Precision	100.00	100.0

Figure 6: Implementation summary table for SVM models showing Accuracy, AUC, and Precision

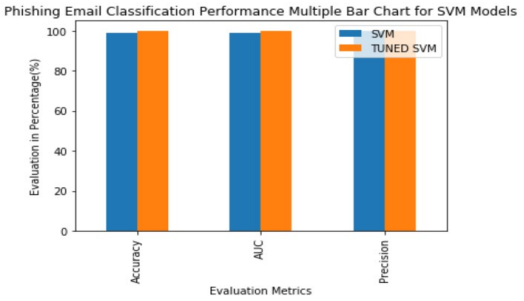

Figure 7: SVM models comparison using Accuracy, AUC, and Precision scores

For experiment one, base SVM model recorded an Accuracy of 99.83%, an AUC score of 99.83%, and a Precision score of 100%. Also, the execution time for training the model and testing the model were 3.37s and 1.14s respectively.

Experiment two shows that all metrics in the optimized SVM model i.e., Accuracy, AUC and Precision had perfect scores of 100%. Also, the execution time for training the model and testing the model were 133.43s and 0.21s respectively.

An analysis of the performance of the two models, for experiment one with an Accuracy of 98.86% and a Precision of 100% suggest a type II error. This means that the base model of SVM wrongly classified some phishing email as legitimate email. On the other hand, experiment two, the model perfect classifies all email group correctly.

4.3.2 Random Forest (RFC) Implementation, Analysis and Result

Random Forest is yet another popular Supervised Learning algorithm used for classification problems also. The algorithm is composed of many decision trees (hence the "forest" is attached to the algorithm name) as an ensemble which is trained with using the bagging technique. The general idea of the bagging technique is that the combination of learning models increases the overall performance.

4.3.2.1 Implementation

Two sets of experiments were carried out in this implementation. More details will be included in the configuration manual.

4.3.2.2 Result and Analysis

	RFC	TUNED RFC
Accuracy	100.0	100.0
AUC	100.0	100.0
Precision	100.0	100.0

Figure 8: Implementation summary table for RFC models showing Accuracy, AUC, and Precision

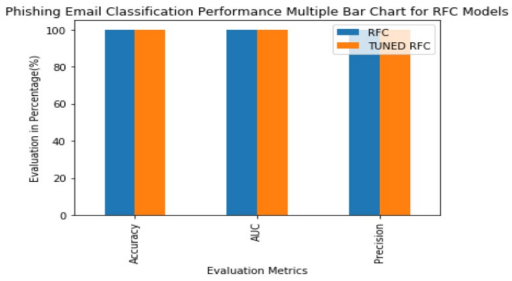

Figure 9: RFC models comparison using Accuracy, AUC, and Precision scores

Figure 8 and figure 9 shows the evaluation of the model's implementation for RandomForest Classifier. The column named 'RFC' is the report for experiment one which is our base model implementation while the column named 'TUNED RFC' is the report for experiment two which is the optimized model of the implementation.

For experiment one, the base REC model recorded an Accuracy of 100%, an AUC score of 100%, and a Precision score of 100%. Also, the execution time for training the model and testing the model were 0.85s and 0.09s respectively.

Experiment two also shows that the optimized RFC model had an Accuracy score of 100%, an AUC score of 100%, and a Precision score of 100%. Also, the execution time for training the model and testing the model were 943.7s and 0.09s respectively.

An analysis of the performance of the two models, experiment one and experiment two implemented models perfectly classify all email groups correctly. There was no type I or type II errors recorded.

4.3.3 Deep Neural Network (DNN) Implementation, Analysis, and Result

A Deep Neural Network is an Artificial Neural Network (ANN) with multiple layers between the input and the output layers. ANN is a collection of nodes called neurons connected synapses that can transmit signal (a signal is a real number which is the output generated by a neuron's non-linear function of the sum of its inputs) to other neurons.

4.3.3.1 Implementation

This implementation uses the Keras Python library by importing the Sequential module, KerasClassifier module, and Dense module to build the DNN. More details will be included in the configuration manual.

4.3.3.2 Result and Analysis

	DNN
Accuracy	100.0
AUC	100.0
Precision	100.0

Figure 10: Implementation summary table for DNN model showing Accuracy, AUC, and Precision

Figure 10 shows the evaluation of the model's implementation for the Deep Neural Network (DNN). From figure 10 above, the DNN model has an accuracy of 100%, an AUC score of 100%, and a Precision score of 100%. The score from the experiment indicates that they correctly classified all emails correctly. No emails were wrongly classified as phishing or legitimate emails is no type I or type II errors were recorded. Also, the execution time for training the model and testing the model were 436.85s and 2.29s respectively.

4.4 Comparison of Implemented Models

	SVM	RFC	DNN
Accuracy	100.0	100.0	100.0
AUC	100.0	100.0	100.0
Precision	100.0	100.0	100.0

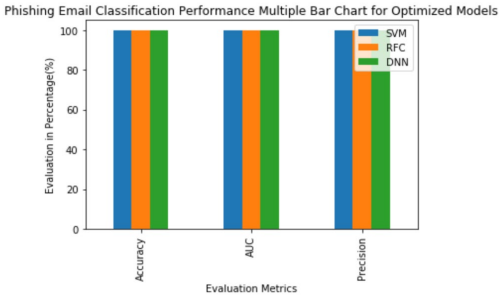

Figure 11: Implementation summary table for all models showing Accuracy, AUC, and Precision

Figure 12: All implemented model's comparison using Accuracy, AUC, and Precision scores

Figure 11 and figure 12 shows the summary table for the performance of all three models (SVM, RFC, and DNN) implemented in classifying emails as either phishing emails or legitimate emails. The 3 models had a 100% accuracy score, 100% AUC score, and 100% Precision score indicating that the implemented models perfectly and correctly classified all email samples as phishing emails or as legitimate emails with no type I or type II error recorded. Based on the time efficiency of the implemented model, SVM used 133.43s in training the model and 0.21s to test the model, RFC used 943.7s in training the model and 0.09s to test the model while DNN used 436.85s in training the model and 2.29s to test the model.

4.5 Comparison of implementation with existing Models

Figure 13 below shows a list of some of the phishing email classification models in the literature review in section 2 alongside this research implementation. This side-by-side presentation is comparing this research implementation to existing models.

Author(s) and Publication	Criteria	Model	Accuracy
Bagui, S., Nandi, D., Bagui, S., and White, R., J. (2021)	Phishing Email Classification	Convolutional Neural Networks (CNN)	96.34%
Fang, Y., Zhang, C., Huang, C., Liu, L., and Yang, Y. (2019)	Phishing Email Detection	Convolutional neural networks (RCNN) model with multilevel vectors	99.85%
Bagui, S., Nandi, D., Bagui, S., and White, R., J. (2021)	Phishing Email Classification	CNN with Word Embedding	96.34%
	Phishing Email Classification		84%

Rastenis, J., Ramanauskaite, S., Suzdalev, I., Tunaityte, K., Janulevicius, J., and Cenys, A. (2021)		Support Vector Machine (SVM)	
Rawal, S., Rawal, B., Shaheen, A., and Malik, S. (2017).	Phishing Email Detection	Support Vector Machine (SVM) and Random Forest	99.87
Yasin, A., and Abuhasan, A. (2016)	Phishing Email Detection	Random Forest Algorithm	99.1%
Ebong, Maurice A. (2022)	**Phishing Email Classification**	**Deep Neural Network (DNN)**	**100%**

Fig 13: Comparison of phishing email classification with the existing model

5 Discussion

Two sets of experiments carried out in this research work aimed at addressing the research question asked in section 1.2. The first experiment was designed as the control experiment (base model experiment) using the default parameter in instantiating the models (SVM and RFC) implemented in classifying phishing emails. To achieve a good result, the dataset was preprocessed and missing data were removed, highly correlated data were removed, the issues posed by an imbalance in the dataset were also removed to create a model that without bias as the model attempts to classify phishing emails. The results obtained from experiment one showed that SVM had an accuracy of 98.86%, an AUC score of 98.85%, and a precision of 100% with a recorded training time. RFC had an accuracy of 100%, an AUC score of 100%, and a precision of 100%. The implemented models were already very good as the random forest model had 100% accuracy.

To implement models that will deliver the best solution every time it is required to classify an email, a second experiment was conducted to enhance the performance of the models implemented. Also, a third model Deep Neural Network (DNN) which is deep learning was implemented to completely address the research question concerning machine learning and deep learning impact and to complete satisfaction all the set objectives of this research work section 1.3. Hyperparameter tuning was applied to the SVM and RFC models to initialize the models with optimal parameters for better performance. The results obtained from experiment two showed that SVM had an accuracy of 100%, an AUC score of 100%, and a precision of 100%, RFC had an

accuracy of 100%, an AUC score of 100%, and a precision of 100% and DNN had an accuracy of 100%, an AUC score of 100%, and a precision of 100%. The results showed that the second experiment was a better model implementation. The dataset used in the implementation was varied in terms of the size of the training and testing with multiple re-runs of the implemented models, the result was the same.

6 Conclusion and Future Work

Using the Natural Language Toolkit (nltk) Python's library, the Nazario email corpus and Enron email dataset was tokenized, stemmed, and stopwords removed to extract meaning and relevant features used in the classification of the phishing email. Feature engineering was applied to the extracted feature from the email corpus used in this research implementation to optimize the performance of the models and for better analysis. They remove the imbalance in the dataset, SMOTE resampling technique was applied to the dataset, and then the grid search which is a hyperparameter tuning approach was applied to the model to enhance their performance. The three models implemented included two machine learning techniques SVM and RFC and a deep learning technique DNN all had 100% accuracy, AUC score, and precision.

In conclusion, deep learning and machine learning techniques, and the NLP greatly impact the classification of phishing using historical data records. The result from the experiment conducted indicates that the research question has been addressed adequately that NLP combined with deep learning techniques or machine learning techniques can impact the classification of phishing emails.

In the future, the dataset used to carry out this research could be updated and used on the implemented model to ascertain that the model can retain 100% efficacy in classifying new phishing techniques as they evolve.

7 Acknowledgment

I would like to give glory and honour to the Almighty God for the abundant blessings he bestowed on me from the beginning of this academic journey till now. I would graciously thank my Project Supervisor, Dr. Vikas Sahni for his patience, tolerance, and mentorship during this period. I would thank all my lecturers at the National College of Ireland. I would like to thank my Godfather, Mr. Godwin Okon for his advice, I would like to thank Very Rev. Father James Kelleher and Very Rev. Father Michael Etim for their prayers and advice and I would also like to thank Mr. /Mrs. Fawzy Umoru and family for their love and care towards me.

Finally, I would like to thank my father and mentor, Engr. Aniefiok S. Ebong for making all these come true, for his guidance, prayers, patience, tolerance, love, and care, I would like to thank my mother, Mrs. Catherine A. Ebong for her love, care, and prayers, and lastly my siblings; Anthony, Martin, Cyril and Danette as they all played important roles to see me accomplish this. May God almighty bless you all abundantly.

8 References

Akinyelu, A. A, and Adewumi, O. A. (2014). Classification of phishing emails using random forest machine learning technique. Journal of Applied Mathematics Volume 2014 http://dx.doi.org/10.1155/2014/425731.

Bagui, S., Nandi, D., Bagui, S., and White, R., J. (2021). Machine learning and deep learning for phishing email classification using one-hot encoding. Journal of Computer Science.

Burita, L., Matoulek, P., Halouzka, K., and Kozak, P. (2021). Analysis of phishing emails. AIMS Electrical and Electronics Engineering Vol.5, Issue 1: 93-116. Doi: 10.3934/electreng.2021006.

Basnet, R., Mukkala, S., and Sung, A., H. (2008). Detection of phishing attacks: A machine learning approach. Soft Computing Applications in Industry, pp. 373-383.

Daniel, A. J., Reshma, G., and Selvarani, C. (2021). International Journal of Advanced Research in Computer and Communication Engineering Vol.10, Issue 7. DOI 10.17148/IJARCCE.2021.10736.

Deshpande, A., Pedamkar, O., Chaudhary, N., and Borde, S. (2021). Detection of phishing websites using machine learning. International Journal of Engineering Research and Technology (IJERT) Vol. 10, Issue 5, ISSN: 2278-0181.

Fang, Y., Zhang, C., Huang, C., Liu, L., and Yang, Y. (2019). Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism. IEEE Access 2169-3536.

Gutierrez, L. F., Abri, F., Armstrong, M., Namin, A., and Jones, K. S. (2020). Phishing detection through email embedding. IEEE International Conference on Big Data.

House, D. (2013). An assessment of user response to phishing attacks: The effect of fear and self-confidence. The University of Texas Arlington.

Kumar, A., Chatterjee, J. M., and Diaz, V. G. (2019). A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing. International Journal of Electrical and Computer Engineering (IJECE) Vol. 10, No. 1, pp. 486-493, ISSN 2088-8708.

Lansley, M., Kapetanakis, S., Polatidis, N. (2020). SEADer++v2: Detecting social engineering attacks using natural language processing and machine learning. IEEE.

Mahajan, R., and Siddavatam, I. (2018). Phishing website detection using machine learning algorithms. International Journal of Computer Applications (0975-8887) Vol. 181, No.23.

Ozcan, A., Catal, C., Donmez, E., and Senturk, B. (2021). A hybrid DNN-LSTM model for detecting phishing URLs. Neural Computing and Applications.

Rastenis, J., Ramanauskaite, S., Suzdalev, I., Tunaityte, K., Janulevicius, J., and Cenys, A. (2021). Multi-Language spam/phishing classification by email body text: Toward automated security incident investigation. 10,668. https://doi.org/10.3390/electronics10060668

Rawal, S., Rawal, B., Shaheen, A., and Malik, S. (2017). Phishing detection in emails using machine learning. International Journal of Applied Information Systems (IJAIS)—ISSN: 2249-0868 Foundation of Computer Science FCS, New York, USA.

Salloum, S., Gaber, T., Vadera, S. and Shaalan, K. (2021). Phishing email detection using natural language processing techniques: A literature review. 5th International Conference on AI in Computational Linguistics. Procedia Computer Science 189 (2021) 19-28.

Senturk, S., and Yerli, E. (2017). Email phishing detection and prevention by using data mining techniques. 2nd International Conference on Computer Science and Engineering.

Shekokar, N. M., Shah, C., Mahajan, M., and Rachh, S. (2015). An ideal approach for detection and prevention of phishing attacks. Procedia Computer Science 49 (2015) 82-91.

Siddique, Z, B., Khan, M, A., Din, I, U., Almogren, A., Mohiuddin, I., and Nazir, S. (2021). Machine learning-based detection of spam emails. Hindawi Scientific Programming Volume 2021.

Sonowal, G. (2020). Phishing email detection based on binary search feature selection. SN Computer Science 1: 191 https://doi.org/10.1007/s42979-020-00194-z

Sujithra, T., Dwivedi, N., Utakarsha, A. (2020). Detection of phishing websites using deep learning and machine learning. Journal of Critical Review, Vol. 7, Issue 8, ISSN- 2394-5125.

Verma, R., and Hossain, N. (2014). Semantic feature selection for text with application to phishing email detection. Department of Computer Science, University of Houston.

Yasin, A., and Abuhasan, A. (2016). An intelligent classification model for phishing email detection. International Journal of Network Security & Its Application (IJNSA) Vol.8, No.4.

Zaladavia, F., Pandey, S., Pachpande, P., Nevrekar, A., Govilkar, S. (2020). Detecting phishing attacks using natural language processing and deep learning models. International Journal of Creative Research Thoughts (IJCRT) Vol. 8, Issue 5, ISSN: 2320-2882.

Zhang, X., Chen, F., and Huang, R. (2018). A combination of RNN and CNN for attention-based relation classification. 8th International Congress of Information and Communication Technology (ICICT). Procedia Computer Science 131, 911-917.