National College of Ireland

Evaluate the use of Supervised Machine Learning Algorithms in the detection of phishing attacks.

MSc Research Project
Cybersecurity

Michael O'Brien
x20191359

School of Computing
National College of Ireland

Supervisor: Ross Spelman

National College of Ireland

Project Submission Sheet – 2021/2022

Student Name:	Michael O'Brien		
Student ID:	x20191359		
Programme:	Cybersecurity	**Year:**	2023
Module:	MSC Research Project		
Lecturer:	Ross Spelman		
Submission Due Date:	01st February 2023		
Project Title:	Evaluate the use of Supervised Machine Learning Algorithms in the detection of phishing attacks.		
Word Count:	12,071 (Excluding Cover Sheet and Project Submission Sheet)		

I hereby certify that the information contained in this (my submission) is information pertaining to 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 references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	Michael O'Brien
Date:	01st February 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

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

Evaluate the use of Supervised Machine Learning Algorithms in the detection of phishing attacks.

Michael O'Brien
x20191359
MSc in Cybersecurity
National College of Ireland

Abstract

The weakest elements in any cybersecurity framework are the end users, people. Hackers have been taking advantage of the end users since the birth of the modern Technology age. Social Engineering, in particular phishing is seen as an easy method attack. The Nigerian prince, one of the earliest global phishing attacks, which is still doing the rounds today, is still estimated to be making over $700,000 a year. And it is not because there isn't email Security systems or not because people are aware of these attacks, it's just a human condition, we have a lot on our minds, we are not paying full attention and we make mistakes, it is as simple as that.

In today's world, Artificial Intelligence (AI) and Machine Learning (ML) have been intergraded to millions of systems and within the IT security sector, because of the lack of security professional available to manage an ever-growing sector, AI and ML are one of the, if not the, largest growing technologies being utilized. And when we speak about AI and ML, we are not talking about the T1000 (Terminator 2) or the singularity (Vernor Vinge, 1993) intelligence, we are talking about intelligence demonstrated by machines based on making predictions from Data by using Algorithms.

This research paper is going to examine the threat of phishing in today world. The paper will examine existing techniques used to identifying phishing attacks and phishing URL's. The goal of this research is to evaluate the use of these AI/ML algorithms and their accuracy in the detection of phishing attacks compared to existing detection methods currently being used.

The research will examine three Machine Learning Algorithms (Decision Tree, Random Forest, & Naive Bayes) and evaluate their accuracy in detecting phishing attackers from a known dataset, using supervised learning. The goal from this research paper is going to be data, calculated data that can be used and compared to existing methods of phishing detection. Data that is measurable, tested and can be used for further research. Aiming to identify which Machine Learning algorithm, with supervised learning approach can be used to achieve the most accuracy in the detection of phishing attacks.

Table of Content

1 INTRODUCTION

With humans still being the weakest link in the cybersecurity chain, phishing attacks are seen by cybercriminals as an easy pivot point to access corporate networks. Phishing attacks have been in the top cyber threats for the last number of years, according to a Data Breach Investigations Report (Verizon, 2021), carried out by Verizon Enterprise in 2021, it identified phishing attacks as one of the main causes of the data breaches it analysed, more specifically reporting that phishing was the cause of 36% of breaches in 2021, up from 22% in 2020. In recent research carried out by Tessian (Rosenthal, 2022) in 2021, it reported that each employee will receive 14 malicious emails, on average per year. And when you look at specific attacks, like the HSE ransomware attack in March 2021, a PwC report (PwC, 2021) identified that the opening of an attachment on a phishing email led to the cyberattack, it is clear that continued investigation and research into additional security tools, such as the use of Artificial intelligence (AI), Machine Learning (ML) and Deep Learning (DL), in the defection and prevention of phishing attacks will add substantial value to the cybersecurity field.

1.1 Research question and the proposed goals of research project

This research aims to evaluate the use of Machine Learning Algorithms in detecting phishing attacks. The questioned being examined is - Evaluate the use of Supervised Machine Learning Algorithms in the detection of phishing attacks. The research paper will use the following Machine Learning Algorithms -Random Forest Algorithm, Decision Tree, and Naive Bayes. Using an existing publicly available dataset, the algorithms will be evaluated, using supervised learning techniques. The proposed goal of the research is to achieve statistical outcomes that are measurable, repeatable, and tested. This information can then be analysed with the goal of identifying which of the three algorithms performed best, against defined criteria, in the detection of dangerous URL's used during phishing attacks.

The importance of research into this IT security field can clearly be seen in several recent IT Security reports, such as the recent Proofpoint report (Proofpoint, 2021) carried out in 2021, which details the follow statistics:

- 86% of organisations faced bulk phishing attacks
- 79% of organisations saw attacks targeting specific end users
- 78% of organisations saw email-based ransomware attacks.

The use of AI and ML in the IT security sector has become increasingly popular, with several factors driving this trend, including the lack of security professionals available to manage an ever-growing sector. The outcome of this research will have a significant benefit to the IT security sector by allowing the automation of everyday security tasks. AI and ML can handle simple, every day, and repeated threats such as bulk phishing attacks, thereby allowing cybersecurity professionals more time to focus on planning for a more secure future. The results of this research will help organizations to make more informed decisions when it comes to selecting the most appropriate algorithm that can be used to identify phishing URLs. This will ultimately improve an organisations security posture.

1.2 Artificial intelligence, Machine Learning and Deep Learning

Artificial intelligence (AI) is a field of study that develops applications and/or systems that can perform tasks that generally require human intelligence, these applications and/or systems are programmed to simulate human beings. AI uses many different approaches and techniques, including machine learning and deep learning. So, if you look at AI in the form of a circle like shown in Figure 1, AI is the outer most circle but inside that is a smaller circle called ML and then again within that smaller circle is another circle called DL.

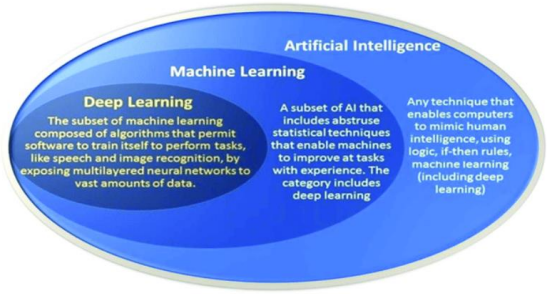

Figure 1: Relationship of AI, ML, and DL

Machine learning (ML) is a subset Artificial intelligence (AI) which involves using statistical modelling and algorithms to allow systems and/or applications to improve how they perform through learning. These algorithms and models are used for the analysis of data and for predicting outcomes or decision making when these predictions and decisions have not been explicitly coded. There are different types of machine learning:

- Supervised Learning.
  This is when a particular model gets trained on a dataset that has already been labelled, basically even line item within the dataset has already been labelled, for example item A – good, item B – Bad. This training dataset is used to train the model to make predictions related new data being modelled. It is trained to predict the presents of something by using a training dataset that contains thousands even millions of items labelled correctly as the required item. It will identify pictures of humans be using a training dataset containing pictures of humans and pictures with no humans correctly labelled as Human or non-human.
- Unsupervised Learning.
  Oppositive to supervised learning, this type of learning is where a model is trained on a unlabelled dataset. There is no required output. This type of model just looks for structure and/or in the dataset. Example of unsupervised learning algorithms are dimensionality reduction and Clustering.
- Reinforcement Learning.
  This type if learning model is based on learning by its mistakes. The model is trained to make decisions based on the outcome of actions taken by it. It is used in tasks such as game playing and robotics.

Then deep learning, this is in turn a subset of machine learning. It involves the training ANN (Artificial Neural Networks) than contain lots of layers. These are also called deep neural networks. These ANN's (Artificial Neural Networks) have used the human brain as inspiration and how the brain can instantaneously identify features from data is sees or absorbs.
Basically, what Artificial Intelligence is trying to achieve is to imitate human intelligence using computer/machines. Statistical Modelling and algorithms are used within Machine Learning to improve its ability and performance. Deep Learning uses deep neural networks. These deep neural networks contain multiple layers for training.
The supervised machine learning algorithms that will be utilised within this research paper are Random Forest Algorithm, Decision Tree Algorithm and Naive Bayes Algorithm. The focus of this research paper is to research how to these algorithms perform at identify Phishing URLs contained within phishing attacks.

1.3 Phishing

Phishing is a form of cybercrime that involves fooling people into giving away personal or corporate data, such as login credentials and/or financial. This is typically done by posing as a trustworthy entity, such as a bank or a government organisation, and sending an email/message that prompts the recipient to click on an embedded URL within the email. This is a type of social engineering. This is where cyber criminals try to trick end users into given away personal or corporate data. This is done by the cyber criminals pretending to be someone they are not like a bank, a government organisation, a friend, a family member, or a colleague. These types of attacks are very serious and can have a huge impact to both people and companies. For the individual end users, it can cause a theft of their PII (Personal Identifiable Information) or impact them financially. For companies, they can cause a loss of corporate information or staff information, data breaches which can result in huge GDPR fines, huge disruptions in operations and reputational damage which may take years to repair. The is multiple types of Phishing attach, such as:

- Email phishing
  This is typically the most used phishing technique because it reaches the end user directly. The cyber criminals will create a phishing email to resemble a legitimate email. They will attempt to have the phishing email appearing as if it was sent by a reliable source like a bank or global recognised company (Amazon). The phishing email will try to take advantage for several human senses, like a sense of urgency or a sense of curiosity or offer a warning of a compromise that requires immediate action. It will have an embedded phishing URL, which will redirect the end user to a phishing website that the cyber criminals have designed to look like a real website.
- Spear phishing
  Like the standard straightforward phishing attack except, instead of being generalised, the cyber criminals create a specific email to attack a particular person, group, or company. The cyber criminals will research their victims, using social engineering techniques, to tailor the attack for the specific person, group, or company that they are attacking.

Information such as CEO names, director names, employer job title, place of work, education, as well as personal information such as name, address, family members, hobbies are all easily available online and this information can be used within a spear phishing attack. This type of attack will often be more successful than standard phishing attacks and therefore must be seen as a very serious threat.

- Whaling
  A Whaling attack is a phishing attack which exclusive attacks executives, such as C level executives, directors' senior managers, etc. These 'Whales' will have higher privileges and/or access to highly important data related to companies. The same techniques used in Phishing and spear phishing are being used. The main goal of the cyber criminals is to fool the executives into giving away personal and corporate data, such as username & passwords, company financial information or confidential business information and this data will be used for financial gain. These attacks are very damaging to companies based who is being targeted and the type of data being phished. The reputational damage to companies who fall victim to a whaling attack will remain for years.
- Smishing
  This is a form of phishing attack that uses mobile instant messages and/or text messages as the point of attack. Like the email form, these phishing messages will appear to be legitimate. They will mostly contain embedded phishing URL but sometimes they will try to open a communication channel to fool end users to send personal data via instant messages and/or text messages.
- Vishing
  This form of phishing attack is an attack using a phone call. An end user will receive a call from a cybercriminal, pretending to be from a legitimate source. The cybercriminal will try and convince the victim to give them PII (Personal Identifiable Information) during the phone call. This is also called voice phishing or phone phishing.
- Angler phishing
  This form of phishing attack take place within social media platforms. Cyber criminals will design social media posts, tweets, etc., using social engineer techniques, to fool end users into interacting with them. These social media posts, tweets, etc. will often contain a phishing URL. Like all the other forms of phishing, the goal is to harvest PII (Personal Identifiable Information) or install malware on the end user device.

All people need to be careful when it comes to all forms of communication. End users should trust but verify, always verify that the communication. If any communication looks for PII (Personal Identifiable Information) or requires authentication to continue, alarm bells should be going off. If end users have any doubt about a received communication, they should verify it via an alternative method of communication or by going to the official website of the organization or entity.

Phishing attacks typically have three steps, see figure 2. Step one, setting up a malicious URL in the form of a fake website, which is designed to mimic an existing, well-known website. This step also includes designing the phishing email. Once the fake website and phishing email are set up, the attacker deploys the attack by sending the email to as many end users, victims, as possible. Step 3, the last step, involves the end user, who is directed to the fake website after clicking the URL link in the email. When some end users become a victim to the phishing attack and enters their credentials, the attacker can then store this data to use later in future attacks or for sale on the dark web. Overall, the objective of a phishing attack is to trick the end user into providing attacks with PII (Personal Identifiable Information) or to install a malicious bit of software on the end point.

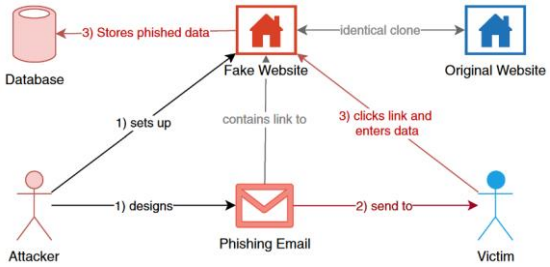

Figure 2: Email as the attack method for a phishing attack

Phishing is a cybercrime that tricks end users into giving away private or sensitive data, PII (Personal Identifiable Information) or corporate information, to the cybercriminals. Protection against phishing, end users and organizations need to ensure that legitimate when clicking on links or entering information into forms and use anti-phishing software. It is key for end users to be aware of the common tactics and techniques used by attackers, and to be careful when receiving emails with embedded links or which as end users to enter personal information. Training on how to spot phishing attempts can also be beneficial, both for individuals and organizations. This can include training on how to identify common phishing tactics, such as using urgency, “Your account will be disabled immediately” or trust “This is your bank”, and on how to verify the authenticity of a request for sensitive information. We are research the benefit of some of these common tactics and techniques within this research paper

1.4 Research paper structure

This research paper, Evaluate the use of Supervised Machine Learning Algorithms in the detection of phishing attacks, is made up of the following sections, introduction, literature review, research methods & specification, research implementation, research evaluation, research conclusion, future research, references, video presentation and acknowledgment. Throughout this research paper the 5 C’s, Clarity, Cogency, Conventionality, Completeness, and Concision, will be adhered to as must as possible.

The research paper has several sections, each section having a separate heading. With section 1, introduction, already being covered to this point of the research paper, the remainder will cover the remaining 9 sections (literature review, research methods & specification, research implementation, research evaluation, research conclusion, future research, references, video presentation and

acknowledgment). Within the second section, Literature Review, existing academic research which is connected to this research topic is reviewed. Relevant findings/outputs from these existing academic research papers reviewed, compared, and analysed. The Literature Review is used as a steppingstone, help to add value to existing research carried out in this field of study. Within the section titled Methods and Specifications, section 3, we define the methods used and the specification required throughout the entire research paper. Within the 4th Section, Research Implementation, this is one of the key parts to the paper. Within this section we detail what Dataset is being used, the design of the Research lab, what Python Libraries are going to be used, define the Machine Learning Algorithms being used, explain Feature Engineering approaches being used and define Model Training for the research paper. The research in implemented in this section and therefore results will be calculated and available. The 5th section, Research Evaluation. In this section the calculated results achieved in section 4 are evaluated. Several evaluation methods will be used, such as Precision, Recall, F measure, ROC area, False Positive rate, False negative rate, and Accuracy. The 6th Section, Research Conclusion, this is after are all the implementation and evaluation has been carried out, an academic research conclusion will be formed based of the research results achieved. The 7th section, Future Research, will describe possible additional research that could be carried out on the foundation of this research paper. Within the 8th section, references, all academic research papers, websites, books, articles, reports used during the research paper are cited. The 9th section is a recorded step by step presentation related to the research paper to help reviewers of the paper get a better understanding of the approach. And then the 10th Section, acknowledgements, is we all people who helped to complete this research paper are acknowledged.

2 LITERATURE REVIEW

The literature review is an essential part of all research projects being carried out. It develops an understanding of the existing knowledge, theories, and studies related to the area or field being researched. In Section II, we are going to look at related work and critically analysis what this related work has already contributed to the area. This section is going to be used as the jump off point, so that the research carried out during this research might add additional value to the field, to quote Sir Isaac Newton "If I have seen further, it is by standing on the shoulders of giants.". The 7 steps to producing a literature review can be seen in Figure 3.

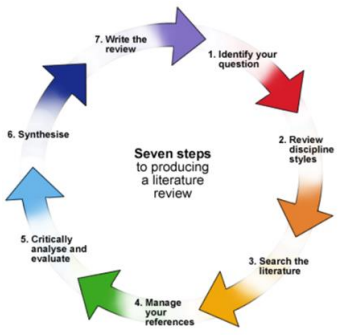

Figure 3: Producing a Literature review

The identification of phishing attacks is a very significant field of research due to the high number of statistics and facts, some already discussed in this paper, supporting how important it is. This together with the known problem, end users are the weakness link and therefore the easiest target for Phishing attacks. Although there is a vast amount of valid research into the subject of Phishing, this paper will concentrate on the use of Machine Learning algorithms in identifying phishing URLs. This section of the research paper use three sections, each section looking at a separate area of research into Phishing attacks. For each research paper reviewed, there will be a summary for the methods used and the findings of the research.

The purpose this literature review is to (i) Show research skills, such as critically evaluate existing research in terms of trustworthiness, value, and relevance, as well as the ability to learn from and build upon previous research, (ii) demonstrate an in-depth understanding of the research topic and how it fits into and adds to the existing body of knowledge related to this field and (iii) achieved a comprehensive understanding of the research topic and its place within the field. The 3 sections this literature review has been broken into are:

- Review of existng research and papers on the current rising threat of phishing
- Review of existing research and papers on traditional methods of preventing phishing attacks
- Review of existing research and papers related to AI and/or ML methods for identifying email related phishing attacks

2.1 Review of the existing rising threat of Phishing

There is plenty of research papers and reports related to the ever-growing threat of Phishing, each one detailing scientific data related to this even increasing global threat. In a report by Tessian (Rosenthal, 2022) which was carried out in 2021, data shows that on average every employee is currently receiving 14 Phishing email a year. The report breaks the data down per sector, and it shows that the retail sector is heavily targeted by phishing attacks, with employees in this sector receiving 49 a year on average. In a report carried out by IBM (Security, 2022), the results show that Phishing is being used more and more as an attack vector to attack organizations, with an increase of 8% reported between 2020 and 2021. This data can be seen in Figure 4.

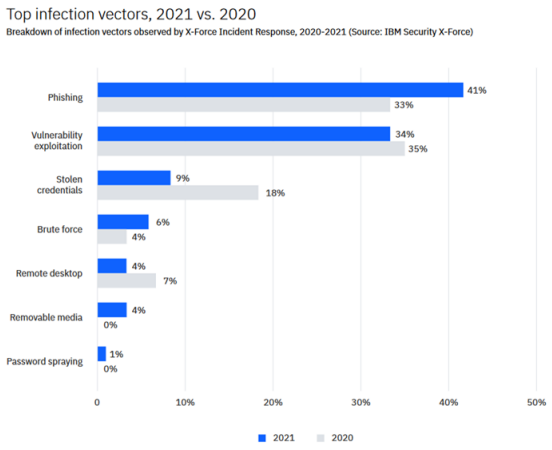

Figure 4: Top infection Vectors (Security, 2022)

More research carried out in 2021, also detailing the growing trend in Phishing attacks, can be seen in ESET's research (ESET, 2021). This research details that malicious email as a form of attack increased by 7.3% in 4 months (May 2021 – August 2021). This trend continues to be report in addition research, such as research carried out by CISCO (CISCO, 2021). Its data shows that in approx. 86% of the companies taken part in the research have had at least one confirmed Phishing URL clicked by an employee. This report also shows that most data breaches, 90%, are the result of a phishing URL. Common themes are used to mask Phishing attacks, such as holidays and report shows that attacks increase by 52% in December (Christmas).

A specific group, known as the Anti-Phishing Working Group (APWG), was formed in 2003. The founder was a man called David Jevans. This group now has over 3200 members, from approx. 1700 global companies. It is an international working group's goal is to reduce or even eliminate fraud, breaches, theft, etc. which are the result of Phishing attacks or Phishing related attacks. The group hopes to achieve this by getting organizations to work together to fight back against Phishing attacks. The group's members are not just small organizations, some global IT Security companies are registered members, such as Kaspersky, Symantec and McAfee, as well as some global financial firms, such as Visa and Mastercard. The group publish reports, regularly. These reports show global statistics related to Phishing. The reports are called the Phishing Activity Trends Reports. These reports have been published since the group was founded, initially annually but changing to quarterly in 2008. The Q4 2021 trends can be seen in Figure 5.

- During the month of December 2021, there was 316,747 recorded attacks. The was the hightest value for a month since APWG began reporting.
- Total number of phishing attacks in Q4 2021 was 3 times that recorded in Q1 2020.
- Just over one fifth (23.2%) of phishing attacks target the Financial Sector (Highest)
- Ransomare type increasing, with Q4 2021 seeing a 36% increase from Q3 2021
- Phishing types
  - Over half (51.8%) are credential theft based
  - Over one third (38.6%) are response based
  - Just under one tenth (9.6%) are malware based.

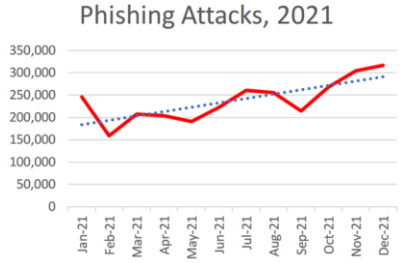

Figure 5: Phishing attack trends 2021 ((APWG), 2021)

2.2 Review of traditional methods of identify phishing emails

There is nothing new about Phishing. It has been used as an attach vector from the very beginning of the internet. Everybody is aware of the Nigerian Prince Phishing, which surprising is as successful today at tricking end users as it was when it was first launched. This part of the

literature review is going to examine the traditional tools and methods used to identify Phishing URLs.

- Browser based plugins and toolbars

  IT Security browser-based toolbars and plugins work on the same principle as signature-based AV solutions. The use known lists, such as whitelists and blacklist to prevent end users from navigating to malicious URLs. The major risk with this approach is new, unknown malicious URLs that have not been added to these known blacklists, and this is exactly what cybercriminals try to take advantage of. Some of these toolbars and plugins update their blacklists once a day, which leave 24hrs for the cybercriminal to cause damage. In research carried out by A.K. Jain and B.B. Gupta (Gupta & Jain, 2016), automatic or dynamic updating of these whitelists/blacklists is investigated to reduce the Zero Day risk. True Positive Rate of 86.02% and a False Negative Rate of 1.48% were achieved in their research.

  In a paper produced by H. Sharma, E. Meenakshi and S. K. Bhatia (Sharma, et al., 2017), called "A comparative analysis and awareness survey of phishing detection tools" examines multiple existing tools used for the detection of Phishing URLs. This paper was presented during a conference in 2017. The dataset used during testing contained 2500 URLs, four fifths of these were malicious URLs. The research calculated the accuracy achieved by all these phishing detection tools, using statistical equations and presented their findings on a table, see figure 6.

Tool	Accuracy
Netcraft	77.12%
Anti-Phishing	94.32%
PhishDetector	58.48%
SpoofGuard	82.16%
BitDefender Traffic Light	93.92%
URLcheck Info	85.72%
LinkExtend	81.12%
SafePreview	65.68%

Figure 6: Overall accuracy of each tool (Sharma, et al., 2017)

- DNS resolution service

  DNS is a service which links a URL/Domain to a specific IP. End users type URLs, this word-based naming convention is easier for end users, but computer systems can easily use Internet Protocols (IP's). There are several types of attacks that can be carried out on DNS, such as Pharming attacks and DNS Poisoning. In research carried out by K. Gajera, M. Jangid, P. Mehta and J. Mittal (Gajera, et al., 2019), they used ANN (Artificial Neural Network) including a Pharming detection technique as part of their proposed method of identifying Phishing attacks. Their research results achieved an accuracy score of 98.77% based on a dataset which contained 4806 URLs.

  DNS Poisoning is another DNS attack method. This is where the attack attempts corrupt, or poison, the DNS connection URL/Domain and IP, so that when the end

user types the URL/Domain and tries to browse to the website, the DNS Poisoning redirects the end user to a malicious website instead of the legitimate. These malicious wed sites are often designed to look and feel like the legitimate website.

- Cryptography/Digital Signatures

  In a research paper carried out by V. Kumar and R. Kumar (Kumar & Kumar, 2015) in 2015, Visual Cryptography is used as a method to identify Phishing URLs. This method is achieved by the encryption of one image in to 2 shares, 1 & 2. The server side of a web site stores one part, share 2, and the client side stores the other sided, Share 1. This sharing happens on the first visit to the web site. On subsequent visits to the web site, both shares are combined to decrypt the image, therefore confirming the web site is the original. If the image is not correctly decrypted the web site is deemed as malicious. A graphically view of this can be seen in figure 7.

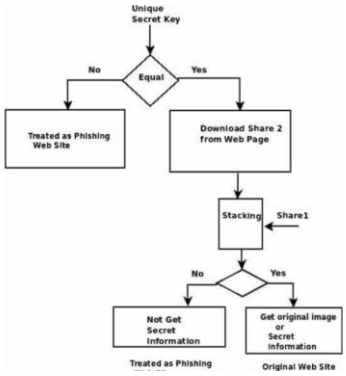

Figure 7: Overall accuracy of each tool (Kumar & Kumar, 2015)

- Awareness training.

  Awareness training of end users is a easy and not technical method of protecting organisations against the threat of Phishing. In research carried out by A. Carella, M. Kotsoev, and T. M. Truta in their 2017 paper (Carella, et al., 2017), end users were brokening into three groups, group one received no training, group two received training documentation and group three received face to face training by a specialist trainer. Click Through Rates simulated Phish exercises were examined over 7 weeks. The results showed that the group trainined by documentation performed best, their Click Through Rates reducing ever week. Group one results remained approximately the same. And group three results dropped immediately after the face to face training of week 1 but gradually increased during the following weeks. Their data can be seen in see figure 8.

Week	Group A	Group B	Group C
Wave 1	56% CTR	50% CTR	52% CTR
Wave 2	54% CTR	44% CTR	36% CTR
Wave 3	52% CTR	32% CTR	34% CTR
Wave 4	48% CTR	22% CTR	38% CTR
Wave 5	48% CTR	16% CTR	40% CTR
Wave 6	52% CTR	10% CTR	48% CTR
Wave 7	54% CTR	8% CTR	50% CTR

Figure 8: Click Through Rates (CTR) for all groups (Carella, et al., 2017)

2.3 Review of the use of AI/ML for the identification of email phishing attacks

This part of the literature review will be looking at existing Artificial intelligence and machine learning research. Existing research of methods and tools already making use of AI/ML to detect Phishing URLs are going to be reviewed. The strengths and weaknesses of the methods and tools will be examined, with the aim of using this existing previous research to add value and benefit this research paper.

In a research paper by A. Basit, M. Zafar, A. R. Javed, and Z. Jalil (Basit, et al., 2020), the use of two classifiers. They proposed using Random Forest classifier as a base and combining it with another classifier. The secondary classifiers used were C4.5 (Decision Tree), ANN (Artificial Neural Network) and KNN (K-Nearest Neighbors). A dataset of 11,055 URLs (6,157 known Phishing) was used.

	ANN+RFC	K-NN+RFC	C4.5+RFC
TP Rate	0.981	0.983	0.977
FP Rate	0.041	0.038	0.054
Precision	0.968	0.970	0.958
Recall	0.981	0.983	0.977
ROC Area	0.997	0.996	0.996
F-measure	0.975	0.976	0.976
Accuracy	97.16	97.33	96.36

Figure 9: Proposed model results (Basit, et al., 2020).

The outcomes of their testing was evaluated against each other, as well as defined research criteria, like ROC Area. See figure 9 above for details. The calculated results of their 2 classifier model showed an improvement when compared to existing methods used.

A research paper carried out by by A. Subasi, E. Molah, F. Almkallawi and T. J. Chaudhery (Subasi, et al., 2017), an Machine Learning algorithm, Random Forest, was modelled to detect Phishing URLs. This algorithm was compared with three other algorithms, C4.5 (Decision Tree), ANN (Artificial Neural Network) and KNN (K-Nearest Neighbors) during the research. The result of their research was that Random Forest performed better that the other three in terms of accuracy, with an accuracy score of 97.36% achieved. And when comparing the papers accuracy score of 97.36% to the best accuracy score of 97.33% achieved by A. Basit, M. Zafar, A. R. Javed, and Z. Jalil (Basit, et al., 2020) research, we can see a small improvement.

Research carried out by T. Peng, I. Harris, and Y. Sawa (Peng, et al., 2018) uses Machine Learning with Natural Language Processing to detect Phishing URLs. This approach is slightly different when compared to existing cited research because this approach analyses the language used within the emails to identify some key malicious indicators of compromise, Figure 11. During their research they developed a new application, SEAHound, using this approach. A 95% accuracy was achieved by SEAHound, not quite as good as the commercially available Netcraft in terms of accuracy but SEAHound scored 13% better than Netcraft with Recall rate, achieving 91%.

In research, performed by A. Karim, S. Azam, B. Shanmugam, K. Kannoorpatti, and M. Alazab (Karim, et al., 2019), Artificial Intelligence and Machine Learning are tested for the detection of spam emails which contain phishing URLs. The examined existing research carried out in this field and generally a table, based on the accuracy of the algorithm, to compare their findings. Their research shown that Artificial Intelligence and Machine Learning systems using one algorithm were

becoming more and more common. Their research also shows that the use or 2 or more algorithms together in system, like A. Basit, M. Zafar, A. R. Javed, and Z. Jalil (Basit, et al., 2020) research, are providing very positive results and merit further research

Long short-term memory (LSTM) model is examined in a research paper carried out by A. C. Bahnsen, E. C. Bohorquez, S. Villegas, J. Vargas, and F. A. González (Bahnsen, et al., 2017). The use LSTM to detect Phishing URLs. During the research LSTM is compared against existing Machine Learning models which use the feature engineering approach to detecting Phishing URLs, such as Random Forest. A. C. Bahnsen, E. C. Bohorquez, S. Villegas, J. Vargas, and F. A. González research shows that LSTM outperformed the existing Machine Learning models using feature engineering. After training their LSTM model, 98.7% accuracy scores were achieved, outperforming Random Forest by 5.2% during the research.

In research carried out by W. Yang, W. Zuo and B. Cui (Yang, et al., 2019), the use a Keyword-Based Convolutional Gated-Recurrent-Unit (GRU) Neural Network. This research tried to identify phishing URLs by using characters as features, text classification features. Phishing URLs take try to exploit known vulnerabilities, such as XSS or SQL injections, by having certain characters as part of the URL. Some of these malicious characters or keywords can be seen in Figure 10. This research achieved a very high accuracy score, 99.6% and above. Much better than what was achieved in similar cited research by A. C. Bahnsen, E. C. Bohorquez, S. Villegas, J. Vargas, and F. A. González (Bahnsen, et al., 2017). The research was carried out on a huge dataset which contained 407212 malicious URL's. The research used a known keyword (malicious Keyword) approach, using feature acquisition. With accuracy scores of greater that 99.6% being achieved, this approach clearly merits further research.

URL sttack type	Malicious keyword
SQL Injection	and, or, xp_, substr, utl, benchmark, shutdown, hex, sqlmap, md5, hex, select, union, drop, delect, concat, orderby, exec
XSS Attack	script, iframe, eval, prompt, alert, javascript, cookie, onclick, Onerror, prompt
Sensitive File Attack	access_log, text/plain, phpinfo, proc/self/cmdline, /fckeditor/
Directory Travel	../, /, ..\, \
Normal	base64, wget, curl, redict, upload, ping, shal, java.lang

Figure 10: Known Malicious Keywords (Yang, et al., 2019)

In a research paper carried out by E. Benavides, W. Fuertes, S. Sanchez, and M. Sanche Deep Learning Models were used, as apposed to Machine Learning Models, for identifying Phishing URLs within the published research (Benavides, et al., 2020). Their research asks the question "What are the techniques of Deep Learning that are currently used in primary studies and how do researchers use these Deep Learning techniques, in order to mitigate phishing attacks?". Within the research paper they detail how they classified the Deep Learning techniques, this can be seen in Figure 11.

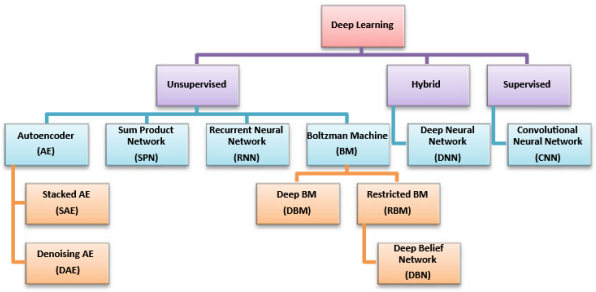

Figure 11: Classification of deep learning techniques (Benavides, et al., 2020)

Their research paper identified 59 existing studies carried out in the field and after analysing these 59 existing studies, 19 studies were identified to be relevant for their research paper. From all the data collected from these 19 relevant studies, their created an easy to read table with all the main features, applied methodology, used algorithms, techniques used and phishing objectives.

In research carried out by P. Yang, G. Zhao, and P. Zeng (Yang, et al., 2019), they also use a Deep Learning approach and develop a twostep approach to identifying Phishing URLs. In their research, step one is to extract character features for the URLs and classify these features using Deep Learning. Step two then uses the URLs webpage features (Code and Text) and statistical features, combines these with the already character features, identified by Deep Learning in the first step. The research paper develops a framework, called MFPD. This framework uses four definitions, these are DCDA, CNN-LSTM, Multi-Dimensional Features and Character Embedding of ui. The Deep Learning approach achieved a accuracy score of 98.99%. The research also recorded that the detection time was quicker. The dataset used contained millions of URLs.

3 RESEARCH METHODS & SPECIFICATION

Section 3, research methods and specification, the techniques and procedures used to carry out the research are examined. These techniques and procedures allow for the gathering and analyse of data to achieve answers to the research question. There are several research methods, such as experimental, observational, survey, etc. The goals of the study and the type of data that is being collected will be dictate the research methods and specification being used.

Specification refers to the detailed description of the research methods and procedures that will be used throughout the research. This includes information on the datasets being utilised, the details of where the datasets were collected, testing lad design and what will the output from the research be evaluated against. Specification is an important step in the research process as it ensures that the study is conducted in a consistent and reliable manner, and that the research can be replicated by other. Research methods and specification are crucial for understanding and interpreting the outputs for the research.

Phishing is becoming an increasingly prominent threat, largely due to its increasing use as an attack vector by cyber criminals. As a result, there is a growing body of research into the use of AI and ML in detecting phishing attacks. Traditional methods are no longer seen as the only defence, with more and more new detection tools being developed using AI and ML elements. This research paper aims to determine which Machine Learning algorithms, Random Forest Algorithm, Decision Tree, and Naive Bayes, approach can achieve the highest

accuracy in detecting phishing attacks for a given dataset. The research paper will evaluate the use of these algorithms, Random Forest, Decision Tree, and Naive Bayes, in detecting phishing attacks.

- Random Forest Algorithm

  Random Forest Algorithms are used in relation to regression and classification tasks. Random Forest is a supervised learning algorithm, and it is also an ensemble machine learning algorithm. It is made up of a group of decision trees, each which are trained on a subset of the data being used. During prediction, it will take an average of the predictions made by each decision tree, which results in a more robust and accurate prediction. Random Forest also allows for feature importance calculation, which can be used to identify the most important features in the data. It is considered as one of the most accurate and robust algorithms in machine learning. See figure 12 for a graphically view of the Random Forest Algorithm

- Decision Tree

  Decision Tree Algorithms are used for classification and regression. It is a non parametric supervised learning method. The decision tree looks like flow chart tree structure, where each feature is represented by a particular internal node. In turn, every decision required is represented by a particular branch, and every result is represented by a particular leaf node. The primary or first node on the decision tree structure is called the Root or Root. During the Decision Tree process, the algorithm learns to divide the data into groups or subsets based on its attribute value. It repeatedly splits the subset into smaller subsets and it makes a decision based on the feature that best splits the data. The final results are represented in the form of tree-like model, where the internal nodes are tests being carried out on a query, the branches of the tree structure are results of tests, and the leaves of the tree structure are the class labels. The Tree like structure of the Decision Tree Algorithm can be seen in figure 13.

- Naive Bayes

  This algorithm can be defined as a probabilistic algorithm. It uses a particular theorem, called Bayes theorem. This theorem details that the probability of an event occurring is equal to the prior probability of the event multiplied by the likelihood of the event occurring during a defined condition. It is mainly used for the task classification. The aim of the algorithm is the prediction of the class or category that is being observed, by using features and/or attributes. The word, 'naïve', relates to the assumption that all the features being used are unique from the other features being used. The problem is that this does not often occur in real data. This aside, this algorithm will perform very well and can be used during spam filtering and text classification. The algorithm is detailed in figure 14.

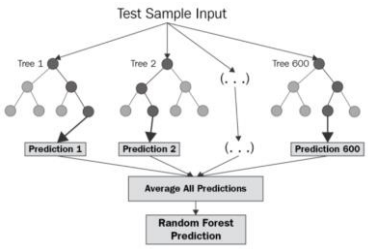

Figure 12: Random Forest Algorithm

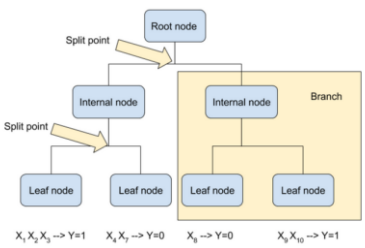

Figure 13: Decision Tree

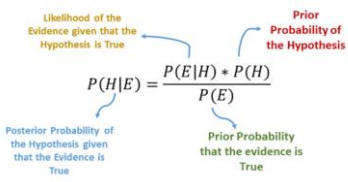

Figure 14: Naive Bayes

During the research, these three algorithms are going to be tested against defined criteria. The testing will be carried out on an existing available dataset, with some alterations to make the data unique. One of the main aims of the research paper is to analyse the algorithms and achieve results that are tested, measurable and repeatable. The goal of this research is to show which of the three algorithms are the best performer at detecting phishing URLs, using predefined criteria as a base for the comparison. This research paper with utilised tables and/or graphs as aids to detail which algorithm is the best performer and which algorithm is the worst performer. Within just section of the research paper the research methods and specifications used to achieve these aims/goals/results will be identified and defined.

3.1 Research Resources and design

From a high level, the Phishing URL dataset will be process by the machine learning algorithm with the expected outcomes, the predefined research criteria, becoming the data to be analysed. The processing element will be carried out within a testing lab. This high-level description is shown by Figure 15 below.

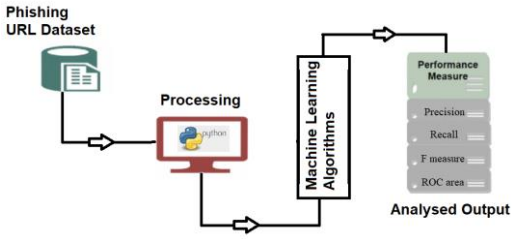

Figure 15: Analysis and Design Diagram

- The Phishing URL Data:

  The Phishing URL Data will be an existing available dataset, available publicly. There will be alterations carried out on the data to make new unique dataset for this research paper.

- The testing Lab

  The testing lab used during this research is a personal PC, with suitable specifications required to complete the research. All required applications will be loaded on the personal PC before testing begins.

- Machine Learning Algorithms

  The identified machine learning algorithms for this research project are:

  - Random Forest
  - Decision Tree
  - Naive Bayes

- Analysed Output

  The outcomes of the research will be analysed against the predefined research criteria (Accuracy, False Negative Rate, ROC Area, F-Measure, Recall & Precision).

3.2 Analysis and Evaluation

The dataset will be split into 2 sections, 1 section containing 30% of the data will be utilised for testing, with the remaining 70% of the data being utilised for the evaluation of the data. This 70%

of the data will be evaluated against predefined statistical criteria. These are Accuracy, False Negative Rate, ROC Area, F-Measure, Recall & Precision.

To understand the analyse the outcomes, researchers must have a strong grasp of the required parameters and the statistical formulas that are being used in the calculation of results before analysing the results.

- False Positive (FP)

  This is when a test incorrectly indicates the presence of a certain condition. For example, during testing if a good URL is identified as a bad or dangerous URL. False positives can lead to unnecessary incident responses and treatments.

- False Negative (FN)

  This value is the result when a test incorrectly shows that something is not present, when it is present. For example, dangerous phishing URL being classified as safe. False negatives can cause huge damage to organisations.

- True Positive (TP)

  This is when a test result that correctly identifies a positive condition. In other words, when a dangerous URL is correctly identified as a dangerous URL, this is a true positive. This is how cybersecurity tools are expected to behave.

- True Negative (TN)

  A true positive is a test result that correctly identifies a negative condition. In other words, a True Negative is when a test correctly shows the presence of something. For example, a safe legitimate URL getting identified as a safe legitimate URL.

To achieve the research papers criteria of calculating and analysing the machine learning algorithms, criteria such as Accuracy, FP (False Positive), FN (False Negative), F-Measure, Recall, Precision and ROC area, the about values (FN, TP, TN & FP) must be used as inputs to calculate the values. Now that the inputs have been detailed and explained, the research criteria must be explained, and formulas used must be detailed.

- Precision

  Precision is the ability of a test to correctly identify true positives, therefore correctly identifying a positive result, among all positive results identified. It is often represented as a proportion or percentage. The precision value is worked out by dividing the TP (True Positives) total value by the sum of the TP (True Positives) and FP (False Positives) total values. A high score indicates that few false positives are present. TP/(TP+FP) is the formula used to calculate this value.

- Recall

  This is the ability for a model to correctly identify all relevant instances or observations. It is the ratio between the number of TP (True Positives) and the total amount of TP (True Positives) plus the total amount of FN (False Negative). When there is a high result, this means that the model being used has a low false negative

rate, therefore it is identifying relevant instances correctly. The formula used is as follows -

TP/(TP+FN)

- F-measure

  This is a statistical measure that uses a combination of both precision and recall when evaluating a binary classifiers performance. F-measure is the harmonic mean of recall and precision. If there is a F-measure, this indicates a good balance between recall and precision. The formula used is as follows -

  2*{(Precision*Recall)/(Precision + Recall)}

- False Positive rate

  False positive rate is a measure that calculates the proportion of false positives out of all negative cases. It tells us the percentage of times that a test incorrectly identifies an individual as having a condition when they do not actually have it. A False Positive can also be called a false alarm or false discovery rate. The formula used is as follows - FP/FP+TN

- False Negative rate

  This is used to show the number of negative results that have actually being identified as positive results. The FN (False Negative) value is achieved by dividing the number of FN (False Negatives) by the number of actual identified negatives. A low score shows that the model is good at identifying negatives, and a high score shows that the model is not good at identifying negatives. The formula used is as follows - FN/FN+TP

- Accuracy

  This is how well a model performs for a given requirement, in other words how good it is at predicting outcomes when given a set of data. In statistics these results are shown as a percentage. Accuracy is worked out by getting the number of correctly predicted outcomes, dividing it by the number of total predictions. This value is often used when evaluating the performance of a particular model being used. And when evaluating the performance multiple different models, it can be used as a good comparison value. But as with all values mentioned here, they should not be looked at in isolation, all calculated results should be looked at, to achieve an overall view of how well a particular model is performing. The formula used is as follows - (TP+TN)/(TP+TN+ FP+FN).

3.3 Ethical Considerations of the Research

Regarding ethics and ethical Considerations of this research paper, all necessary measures will be taken to adhere to current guidelines. The use of AI and ML raises ethical concerns that must be continually identified, discussed, and evaluated throughout the technology's development, implementation, and impact on society, including social, economic, political, and psychological effects. Principles such as Google's AI Principles (Google, 2022) and Asilomar AI Principles (Stapf-Fine, 2018) have been established to address some ethical issues. Additionally, legal

requirements, such as the GDPR (Union, 2018), must also be considered during research. For example, Article 22 of GDPR regulations provide protection for individuals' information regarding automated decision-making and profiling. This research paper will utilize existing, publicly available datasets and will obtain ethical consent if and where necessary.

4 RESEARCH IMPLEMENTATION

This research paper will study the use of supervised learning algorithms, specifically Forest Algorithm, Decision Tree, and Naive Bayes, in detecting phishing attacks. The effectiveness of each algorithm being used in this research project will be measured, ranked, and compared. The outcomes of the analysis will be displayed withing tables and/ or graphs, which will clearly show the best and worst performing algorithm being researched. The research will include various metrics from phishing attacks to determine the best performing algorithm. The methods and requirements for this research paper will be detailed in this section of the research paper.

4.1 Dataset

The dataset being used is the Malicious URLs dataset (Source: Kaggle, Published 2021). This dataset is made up of 651,191 URL's broken into 4 categories:

- Benign (428,103)
- Defacement (94,457)
- Phishing (94,111)
- Malware (32,520)

I have altered this to make it unique for the research project. This new altered dataset is now made up of 651,200 URL's broken into 2 categories:

- Benign (428,112)
- Malicious (223,088)

Dataset Source:

- Original Dataset source: Kaggle.com
- Original Dataset Name: Malicious URLs dataset
- Original Dataset Author: Manu Siddhartha
- Direct URL to data: https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset

4.2 Research lab

A personal PC, MacBook Pro, was used to set up the testing lab for this research

- Internal Processor – 2.6 GHz 6-core Intel Core i7
- Internal Memory – 32 GB 2400 MHz ddr4
- Start-up Disk – Macintosh HD
- Graphics – Intel UHD Graphics 630 1536 MB
- OS – macOS Monterey (Version 12.6)

The computing program Anaconda was loaded. Anaconda simplifies application deployment and package management. It is very good for scientific computing using Python programming language and R computer programming language. There are specific data analytical packages preloaded within this application. Anaconda will work on most Operating Systems, including Mac OS, Windows, and Linux.

Jupyter Notebook is an application within Anaconda. It is a server-client app which is used for editing and running of documents within a web browser. This app runs locally on the Operating System. Therefore, it does not require internet access to function. Alternatively, it can be installed on a remote sever, accessible through an internet connection

4.3 Python Libraries

Libraries used within Python during this research:

- Numpy:

  This python library is the main library used for scientific computing within the Python Language. It can be used to create multiple derived objects and for multi-dimensional array object. Examples of these are matrices and masked arrays. It is also used for multiple routines when operating on arrays. Routines such as, to logical, sorting basic algebra, statistical operations, sorting, etc. (Numpy, 2022).

- Sklearn:

  This is one of the most useful libraries in python in relation to machine learning. This library has plenty of very effect and efficient tools that can be used with in ML (Machine Learning). It can also be used in statistics for the modelling of regression, classification, dimensionality, and clustering (Jain, 2015).

- Pandas:

  This open-sourced library is used to analysing data and manipulating data. It is easy to use, very powerful and is very fast. The Pandas library has been built on top of the Python language (Pandas, 2022).

- matplotlib:

  This library is used for making interactive, static, and animated visualizations within the Python Language. It helps to make impossible tasks possible and helps to make easy tasks even easier. (Matplotlib, 2022).

- os:

  This library is used for programming functionality that requires operating systems (Python, 2022).

- seaborn:

  Based on matplotlib, seaborn can be used for data visualization. It gives uses an interface, which can be used to creating impressive statistical graphics and diagrams (Seaborn, 2022).

- WordCloud:

  Its library is used to detail how frequently an item, words, appear within a given set of data. It will generate an image with the most frequently occurring word being

displayed the largest and the least frequently occurring smallest. Basically, the frequency for the word will dictate the display size of that word. It is all customisable, such as font used, colours, etc. (Pypi, 2022).

- Time

  This library is basically used for any time related requirements within python (Python.org, 2022).

4.4 Machine Learning Algorithms

The identified machine learning algorithms for this research project are:

- Random Forest Algorithm

  Random Forest is an ensemble machine learning algorithm. It is used in relation to regression and classification tasks. It is a supervised learning algorithm. It is made up of a group or decision trees, each which are trained on a subset of the data being used. During prediction, it will take an average of the predictions made by each decision tree, which results in a more robust and accurate prediction. Random Forest also allows for feature importance calculation, which can be used to identify the most important features in the data. It is considered as one of the most accurate and robust algorithms in machine learning.

- Decision Tree

  This algorithm is a non-parametric supervised learning method. It is used for classification and regression. The decision tree is like flow chart tree structure, in which each internal node represents a particular feature, each branch of the tree represents a decision rule, and then each leaf node represents the particular result. The topmost node within the decision tree is called the root node. The algorithm learns to break the data into subsets based on the attribute value. It repeatedly splits the subset into smaller subsets and it makes a decision based on the feature that best splits the data. The final result is represented in the form of tree-like model, where each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. See figure 13 for a graphically view of the Decision Tree Algorithm.

- Naive Bayes

  The Naive Bayes algorithm is defined as a probabilistic algorithm. It uses Bayes theorem, this says that the probability of an event occurring is equal to the prior probability of the event multiplied by the likelihood of the event occurring during a given certain condition. It is mainly used for the task classification. The goal of this is to predict the class or category being observed, by using features and/or attributes. The word, ‘naïve’, relates to the assumption that all the features being used are unique from the other features being used. The problem is that this does not often occur in real data. This aside, this algorithm will perform very well and can be used during spam filtering and text classification.

4.5 Feature Engineering

Feature Engineering is identifying and extracting features in datasets and utilising these features in formats that can be used by ML algorithms. When it comes to identifying some features for URL's, first you need to understand the structure of a URL.

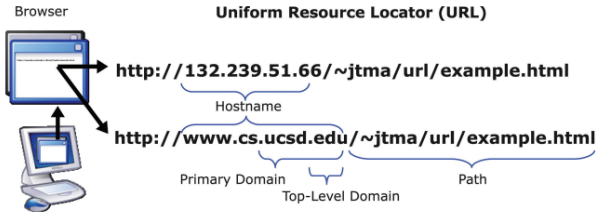

Figure 16: Example of URL structure

For this research project, 22 features have been identified and utilised . Examples of these features are (See Figure 17 for full list of features being utilised):

- Google indexing (google_index):
  Basically, checking to see if the URL has a google index or not.
- URL contains an IP (having_ip_address):
  Cyber criminals are more likely to use IP addresses (IP4, IP6, etc.), so checking if URL's contain IP address will help to identify suspicious URL's.
- Length of top-level domains (tld_length):
  See Figure 16 for details, but in www.example.com, 'com' is the top-level domain. Globally today, must URLs are '.com', within Ireland we also commonly use '.ie'. Therefore, a top-level domain range of between 2 & 3 would help to identify safe URL's.
- Has the URL been shortened (Short_url):
  This checks to see if the URL has been shorted. For example, rb.gy/6oz7zc (www.example.ie) and rb.gy/tkzlby (www.maliciouswebsite.cooom).
- Suspicious words in URL:
  Suspicious keywords like PayPal, login, sign in, bank, account, update, bonus, etc are often found within suspicious URL's and can be used to help identify suspicious URL's.

```
#Predictor Variables being used
X = data[['IP_in_use','abnormal_url', 'google_index', 'count.', 'count-www', 'count@',
       'count_dir', 'count_embed_domian', 'short_url', 'count-https',
       'count-http', 'count%', 'count?', 'count-', 'count=', 'url_length',
       'hostname_length', 'sus_url', 'fd_length', 'tld_length', 'count-digits',
       'count-letters']]
```

Figure 17: Features used within research project

Figure 18 shows the full list of features being used during this research project. These 22 features are used by the Machine Learning algorithms to identify what category the URL is (Safe or malicious).

	url	type	IP_in_use	abnormal_url	google_index	count.	count-www	count@	count_dir	count_embed_domian	...	count-	count=	url_length	hostname_length	sus_url	count-digits	count-letters	fd_length	tld_length	type_code
0	br-icloud.com.br	malicious	0	0	1	2	0	0	0	0	...	1	0	16	0	0	0	13	0	-1	1
1	mp3raid.com/music/krizz_kaliko.html	benign	0	0	1	2	0	0	2	0	...	0	0	35	0	0	1	29	5	-1	0
2	bopsecrets.org/rexroth/cr/1.htm	benign	0	0	1	2	0	0	3	0	...	0	0	31	0	0	1	25	7	-1	0
3	http://www.garage-pirenne.be/index.php?option=...	malicious	0	1	1	3	1	0	1	0	...	1	4	88	21	0	7	63	9	2	1
4	http://adventure-nicaragua.net/index.php?optio...	malicious	0	1	1	2	0	0	1	0	...	1	3	235	23	0	22	199	9	3	1

Figure 18: Dataset with Features

4.6 Model Training

For this research the breakdown will be 30% of the dataset will be used for training and the remaining 70% of the dataset will be used for the evaluation phase, see Figure 19.

```
In [111]: #Training, for this i an using 30% of dataset for training
          X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y, test_size=0.3,shuffle=True, random_state=5)
```

Figure 19: Model Training

5 RESEARCH EVALUATION

This section of the research, research evaluation, this is where the research is assessed in terms of quality and relevance of the work. It involves evaluating the methodology, results, and overall impact of the research carried out, and this is very important for several reasons. Firstly, it will help to ensure that the research is of a very high standard, and it attempts to answer the research question posed. This is important because it helps to ensure that the research being conducted is scientifically valid and can be used to inform decision-making. Secondly, research evaluation helps to identify areas where further research can be carried out.

There are multiple methods that can be used to evaluate research, for example peer review, expert assessment, bibliometric analysis, etc. The main goal of the Research Evaluation is to ensure that the research being conducted is of high quality, is relevant to the field, and can be used to improve our understanding of the field being researched.

For this research, peer review will be utilised. Peer Review, the most common method of evaluating research, involves having other researchers in the same field review a study or utilising existing research in the field to identify the strengths and weaknesses of the research being carried out.

5.1 WordCloud

A word cloud is a visual representation of the most frequently used words from a given dataset. The Python library wordcloud can be used to generate a word cloud image from a collection of text. The images appearance can be customized (font, colour, etc.). In this research paper it will be used to get a better understanding of the use of different keywords (Words, Tokens, Protocol, top level domain, etc.) used within the URL's. We are going to evaluate both, Benign and Malicious individually to get an understanding of the pattern of keywords within each.

- Benign:

  As can be seen in Figure 20, there is large usage of keywords such as wikipedia, wiki, net, com, co uk, org within Benign URL's. These would be indicators of safe URL's.

- Malicious:

  As can be seen in Figure 21, there is large usage of keywords such as com_content, index, php, option, view article within Malicious URL's. These would be commonly known as indicators of malicious URL's.

Figure 20: Wordcount for Benign URL's

Figure 21: Wordcount for Malicious URL's

The WordCloud images (Figure 24 & Figure 25) are very useful within the Feature Engineer section of this research project, as they are used to develop the features utilised within the ML algorithms.

5.2 Feature Distribution

Now to examine the Feature Distribution of both Benign and Malicious URLs from the dataset being researched.

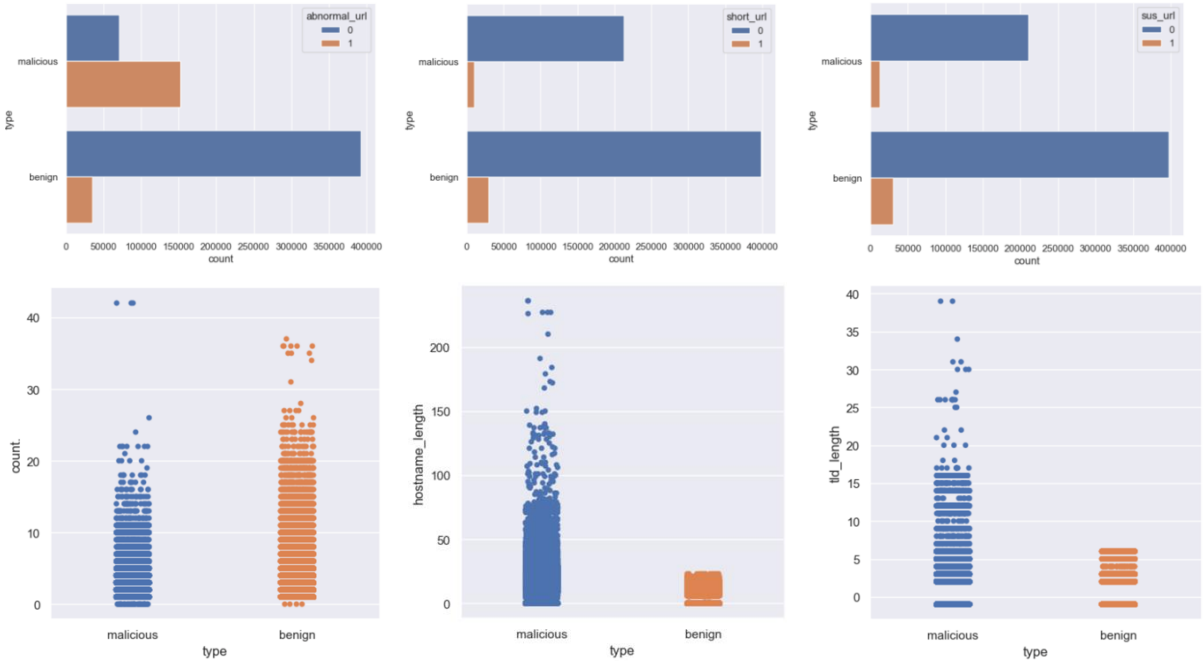

Figure 22: Feature Distribution

In Figure 22 about, the Feature Distribution can be seen of some of the features used during the research. For example, in the feature tld_length (Top Level Domain Length) we see that malicious URL's have much longer Top-Level Domain's, when compared to benign URL's. This is to be expected as most Top-Level Domain's for benign URL's would be '.com', '.net', '.ie', '.co.uk'.
For the hostname_length feature we see the same, where the highest distribution can be seen within malicious URL's, again this is to be expected.
And for feature IP_in_use, the data shows that IPs are in use within the malicious URLs of this dataset.

5.3 Random Forest

This section of the research evaluation we focus on Random Forest Algorithm and running its machine learning model. Figure 23 details the code written for developing the machine learning model for the Random Forest Algorithm.

```
In [166]: #Random Forest Algorithm
          import time
          st = time.process_time()
          import sklearn.metrics as metrics
          from sklearn.ensemble import RandomForestClassifier
          rf = RandomForestClassifier(n_estimators=100,max_features='sqrt')
          rf.fit(X_train,y_train)
          y_pred_rf = rf.predict(X_test)
          print(classification_report(y_test,y_pred_rf,target_names=['benign', 'malicious']))

          score = metrics.accuracy_score(y_test, y_pred_rf)
          print("Accuracy for Random Forest:   %0.3f" % score)

          #instantiate the model
          log_regression = LogisticRegression()

          #define metrics
          y_pred_proba = rf.predict_proba(X_test)[::,1]
          fpr, tpr, _ = metrics.roc_curve(y_test,  y_pred_proba)
          auc = metrics.roc_auc_score(y_test, y_pred_proba)

          #Plotting ROC, including AUC
          plt.plot(fpr,tpr,label="AUC="+str(auc))
          plt.ylabel('True Positive Rate')
          plt.xlabel('False Positive Rate')
          plt.legend(loc=4)
          plt.show()

          #Plot Feature importance during prediction during the model
          feat_importances = pds.Series(rf.feature_importances_, index=X_train.columns)
          feat_importances.sort_values().plot(kind="barh",figsize=(10, 6))

          #Plot CPU Execution Time
          et = time.process_time()
          res = et - st
          print("Execution Time:", res, "Seconds")
```

Figure 23: Random Forest Algorithm

The output from this code can be seen in Figure 24 (Accuracy Score) and Figure 25 (ROC including AUC). We can see from the output data that Random Forest is achieving an accuracy of 97.2%. The AUC value for Random Forest is 0.992 (If predictions are 100% wrong AUC = 0.0 and therefore if predictions are 100% right AUC = 1.0). This data will be compared (to Decision Tree and Naive Bayes) and analysed in Research Conclusion section of this research paper.

```
              precision    recall  f1-score   support

      benign       0.97      0.98      0.98    128434
   malicious       0.97      0.95      0.96     66926

    accuracy                           0.97    195360
   macro avg       0.97      0.97      0.97    195360
weighted avg       0.97      0.97      0.97    195360

Accuracy for Random Forest:   0.972
```

Figure 24: Random Forest Accuracy Score

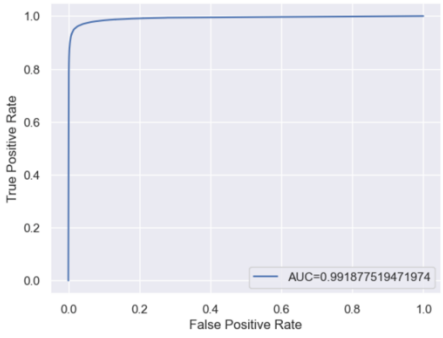

Figure 25: Random Forest ROC

5.4 Decision Tree

This section of the research evaluation we focus on the Decision Tree Algorithm and running its machine learning model. Figure 26 details the code written for developing the machine learning model for the Decision Tree Algorithm.

```
In [167]: #Decision Tree Algorithm
          import time
          st = time.process_time()
          from sklearn.tree import DecisionTreeClassifier
          clf = DecisionTreeClassifier(random_state=0)
          clf.fit(X_train,y_train)
          y_pred_clf = clf.predict(X_test)
          print(classification_report(y_test,y_pred_clf,target_names=['benign', 'malicious']))

          score = metrics.accuracy_score(y_test, y_pred_clf)
          print("Accuracy for Decision Tree:   %0.3f" % score)

          #instantiate the model
          log_regression = LogisticRegression()


          #define metrics
          y_pred_proba = clf.predict_proba(X_test)[::,1]
          fpr, tpr, _ = metrics.roc_curve(y_test,  y_pred_proba)
          auc = metrics.roc_auc_score(y_test, y_pred_proba)

          #Plotting ROC, including AUC
          plt.plot(fpr,tpr,label="AUC="+str(auc))
          plt.ylabel('True Positive Rate')
          plt.xlabel('False Positive Rate')
          plt.legend(loc=4)
          plt.show()

          #Plot Feature importance during prediction during the model
          feat_importances = pds.Series(clf.feature_importances_, index=X_train.columns)
          feat_importances.sort_values().plot(kind="barh",figsize=(10, 6))

          #Plot CPU Execution Time
          et = time.process_time()
          res = et - st
          print("Execution Time:", res, "Seconds")
```

Figure 26: Decision Tree Algorithm

The output from this code can be seen in Figure 27 (Accuracy Score) and Figure 28 (ROC including AUC). We can see from the output data that Decision Tree is achieving an accuracy of 96.6%. The AUC value for Decision Tree is 0.971 (If predictions are 100% wrong AUC = 0.0 and therefore if predictions are 100% right AUC = 1.0). This data will be compared (to Random Forest and Naive Bayes) and analysed in Research Conclusion section of this research paper.

	precision	recall	f1-score	support
benign	0.97	0.98	0.97	128434
malicious	0.96	0.94	0.95	66926
accuracy			0.97	195360
macro avg	0.96	0.96	0.96	195360
weighted avg	0.97	0.97	0.97	195360

Accuracy for Decision Tree: 0.966

Figure 27: Decision Tree Accuracy Score

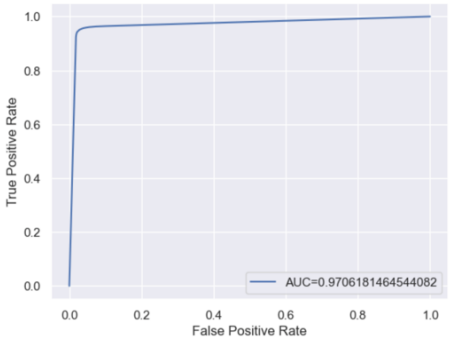

Figure 28: Decision Tree ROC

5.5 Naive Bayes

This section of the research evaluation we focus on the Naive Bayes Algorithm and running its machine learning model. Figure 29 details the code written for developing the machine learning model for the Naive Bayes Algorithm.

```
In [169]: #Naive Bayes Algorithm
          import time
          st = time.process_time()
          from sklearn.naive_bayes import GaussianNB
          gnb = GaussianNB()
          gnb.fit(X_train, y_train)
          y_pred_gnb = gnb.predict(X_test)
          print(classification_report(y_test,y_pred_gnb,target_names=['benign', 'malicious']))

          score = metrics.accuracy_score(y_test, y_pred_gnb)
          print("Accuracy for Naive Bayes:   %0.3f" % score)

          from sklearn.linear_model import LogisticRegression

          #instantiate the model
          log_regression = LogisticRegression()

          #fit the model using the training data
          #log_regression.fit(X_train,y_train)

          #defining metrics
          y_pred_proba = gnb.predict_proba(X_test)[::,1]
          fpr, tpr, _ = metrics.roc_curve(y_test,  y_pred_proba)
          auc = metrics.roc_auc_score(y_test, y_pred_proba)

          #Plotting ROC, including AUC
          plt.plot(fpr,tpr,label="AUC="+str(auc))
          plt.ylabel('True Positive Rate')
          plt.xlabel('False Positive Rate')
          plt.legend(loc=4)
          plt.show()

          #Plot Feature importance during prediction during the model
          #feat_importances = pds.Series(gnb.feature_importances_, index=X_train.columns)
          #feat_importances.sort_values().plot(kind="barh",figsize=(10, 6))

          from sklearn.inspection import permutation_importance
          imps = permutation_importance(gnb, X_test, y_test)
          print(imps.importances_mean)

          #Plot CPU Execution Time
          et = time.process_time()
          res = et - st
          print("Execution Time:", res, "Seconds")
```

Figure 29: Naive Bayes Algorithm

The output from this code can be seen in Figure 30 (Accuracy Score) and Figure 31 (ROC including AUC). We can see from the output data that Naive Bayes is achieving an accuracy of 83.7%. The AUC value for Naive Bayes is 0.895 (If predictions are 100% wrong AUC = 0.0 and therefore if predictions are 100% right AUC = 1.0). This data will be compared (to Random Forest and Decision Tree) and analysed in Research Conclusion section of this research paper.

	precision	recall	f1-score	support
benign	0.84	0.92	0.88	128434
malicious	0.82	0.67	0.74	66926
accuracy			0.84	195360
macro avg	0.83	0.80	0.81	195360
weighted avg	0.84	0.84	0.83	195360

Accuracy for Naive Bayes: 0.837

Figure 30: Naive Bayes Accuracy Score

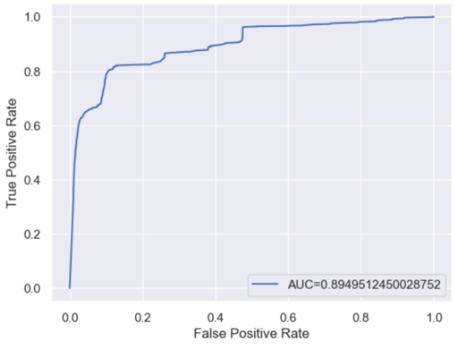

Figure 31: Naive Bayes ROC

6 RESEARCH CONCLUSION

This section of the research, research conclusion, is where the outputs/findings of the research are summarised, and conclusions are formed for the output data that was collected. Therefore, in this section, research conclusion, the output data from Random Forest Algorithm, Decision Tree algorithm, and Naive Bayes algorithm will all be analysed as per the research question identified at the beginning of the research. The first part of the research conclusion will be to analyse these three models individually, one by one.

Firstly, Random Forest Algorithm. During the research evaluation, weighted average values (Weighted Avg) of Precision = 0.97, Recall = 0.97, F1-Score = 0.97 were achieved. This can be seen in Figure 24. The Decision Tree Algorithm matched these values and achieved weighted average values (Weighted Avg) of Precision = 0.97, Recall = 0.97, F1-Score = 0.97. This can be seen in Figure 27. The Naive Bayes Algorithm did not perform as well, with weighted average values (Weighted Avg) of Precision = 0.84, Recall = 0.84, F1-Score = 0.83 being achieved during research evaluation.

But looking closer at the specific data related to three Algorithms (Random Forest, Decision Tree, and Naive Bayes), we see that Random Forest performed very well, with the best individual values for identifying both Malicious and Benign URL's. Benign having values of Precision = 0.97, Recall = 0.98, F1-Score = 0.98. And Malicious having values of Precision = 0.97, Recall = 0.95, F1-Score = 0.96

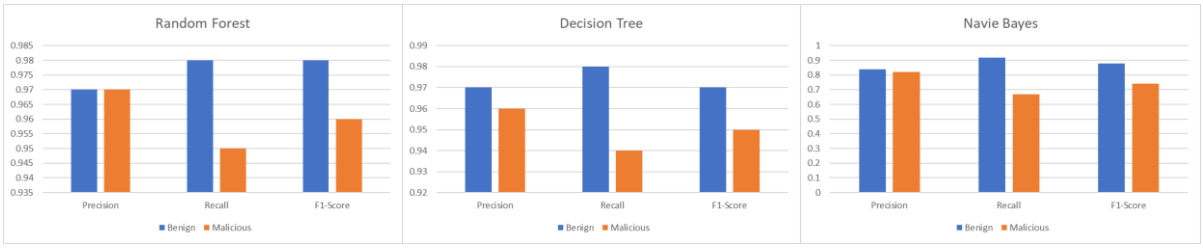

Figure 32: Precision, Recall and F1-Score

Now we can analyse and compare the research evaluation data, for Random Forest Algorithm, Decision Tree algorithm, and Naive Bayes algorithm as a group. When it comes to accuracy, Random Forest algorithm and Decision Tree algorithm performed very well with scores of 97.2% and 96.6% respectively. In terms of accuracy, Naïve Bayes did not perform as well, only achieving a score of 83.7%. In terms on AUC (Range 0-1, if predictions are 100% wrong AUC = 0.0 and if predictions are 100% right AUC = 1.0) we again see Random Forest algorithm and Decision Tree algorithm performed very well with scores of 0.992 and 0.971 respectively. And again, Naïve Bayes algorithm did not perform as well, only achieving a AUC score of 0.895.

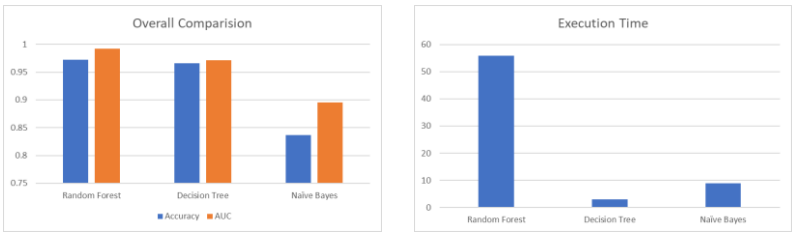

Figure 33: Overall Comparison of Data

But if we are to look at execution time, the Decision Tree Algorithm is the clear winner with an execution time of 2.94 Seconds, followed by Naïve Bayes with an execution time of 8.84 Seconds and finally Random Forest with execution time of a huge 55.84 Seconds.

7 FUTURE RESEARCH

In this section, future research, possible areas and/or fields of future research will be investigated. After competing this research paper, the next logical step is to utilise this data and develop an application, mobile app or browser plugin, using Python models that could take advantage of the findings from this research paper (Shatby, 2022). Developing this application using machine learning will require a full, in-depth, understanding of not only the problem but also a full understanding of the available data, and the techniques, such as programming and Machine Learning, that can be utilised to develop a possible solution. One of the key considerations to bear in mind when developing or designing this application is the training data, in terms of its

availability and accuracy, because phishing emails are constantly evolving, and the features to identify them from legitimate emails are also changing. This means that the data used to train the model must be constantly updated to reflect these changes.

Another consideration is the choice of machine learning algorithm. Even do this research paper focused on Random Forest Algorithm, Decision Tree Algorithm, and Naive Bayes Algorithm, these are not the only Machine Learning models that can be used to detect phishing URLs. As seen during the research paper, each algorithm has its own strengths and weaknesses, and the choice of algorithm will depend on the specific characteristics of the data and level of accuracy required. The application design is also an important consideration. It must be user-friendly and easy to use, as well as able to integrate with existing applications and email systems. The application should be able to provide feedback to users on the likelihood that an email is a phishing attempt, so that they can make informed decisions about whether to open it or not.

The scale of impact of this application and its ability to detect phishing emails would be significant. Phishing is a major problem for both individuals and organizations. A successful application could help to protect individuals and organizations from these attacks and reduce the overall impact of phishing on society.

Taking all this into account, developing an application to detect phishing using machine learning is a challenging task that requires a thorough understanding of the problem, the available data, and the machine learning techniques that can be used to solve it, all of which will lead to Future Research.

8 REFERENCES

Carella, A., Kotsoev , M. & Truta, T. M., 2017. *Impact of security awareness training on phishing click-through rates.* Boston, MA, USA , IEEE.

(APWG), T. A.-P. W. G., 2021. *Phishing Activity Trends Report,* s.l.: APWG.

Bahnsen, A. C. et al., 2017. *Classifying phishing URLs using recurrent neural networks.* Scottsdale, AZ, USA , IEEE.

Basit, A., Zafar, M., Javed , A. R. & Jalil, Z., 2020. *A Novel Ensemble Machine Learning Method to Detect Phishing Attack.* Bahawalpur, Pakistan, IEEE.

Beloglazov, A. & Buyya, R., 2015. Openstack neat: A framework for dynamic and energy-efficient consolidation of virtual machines in openstack clouds. *Concurrency and Computation: Practice and Experience*, 27(5), pp. 1310-1333.

Benavides, E., Fuertes, W., Sanchez , S. & Sanche, M., 2020. Classification of Phishing Attack Solutions by Employing Deep Learning Techniques: A Systematic Literature Review. In: *Developments and Advances in Defense and Security.* s.l.:Springer Nature Singapore Pte Ltd, pp. 51-64.

CISCO, 2021. *Cybersecurity threat trends: phishing, crypto top the list,* s.l.: CISCO.

ESET, 2021. *Threat Report T2 2021,* s.l.: ESET.

Feng, G. & Buyya, R., 2016. Maximum revenue-oriented resource allocation in cloud. *International Journal of Grid and Utility Computing,* 7(1), pp. 12-21.

Gajera, K., Jangid, M., Mehta, P. & Mittal, J., 2019. *A Novel Approach to Detect Phishing Attack Using Artificial Neural Networks Combined with Pharming Detection.* Coimbatore, India, IEEE.

Gomes, D. G., Calheiros, R. N. & Tolosana-Calasanz, R., 2015. Introduction to the special issue on cloud computing: Recent developments and challenging issues. *Computer & Electrical Engineering,* Volume 42, pp. 31-32.

Google, 2022. *Artificial Intelligence at Google: Our Principles.* [Online]
Available at: https://ai.google/principles/
[Accessed April 2022].

Gupta, A. & Jain, B., 2016. A novel approach to protect against phishing attacks at client side using auto-updated white-list.. *EURASIP J. on Info. Security 2016, Article No. 9.*

Jain, K., 2015. *www.analyticsvidhya.com.* [Online]
Available at: https://www.analyticsvidhya.com/blog/2015/01/scikit-learn-python-machine-learning-tool/
[Accessed November 2022].

Karim, A. et al., 2019. *A Comprehensive Survey for Intelligent Spam Email Detection.* s.l., IEEE.

Kumar, V. & Kumar, R., 2015. *Detection of phishing attack using visual cryptography in ad hoc network.* Melmaruvathur, India , IEEE.

Kune, R. et al., 2016. The anatomy of big data computing. *Software—Practice & Experience,* 46(1), pp. 79-105.

Matplotlib, 2022. *https://matplotlib.org/.* [Online]
Available at: https://matplotlib.org/
[Accessed November 2022].

Numpy, 2022. *numpy.org.* [Online]
Available at: https://numpy.org/doc/stable/
[Accessed November 2022].

Pandas, 2022. *https://pandas.pydata.org/.* [Online]
Available at: https://pandas.pydata.org/
[Accessed November 2022].

Peng, T., Harris, I. & Sawa, Y., 2018. *Detecting Phishing Attacks Using Natural Language Processing and Machine Learning.* Laguna Hills, CA, USA, IEEE.

Proofpoint, 2021. *2021 State of the Phish,* s.l.: Proofpoint.

PwC, 2021. *Conti cyber Attack on the HSE,* s.l.: PwC.

Pypi, 2022. *pypi.org.* [Online]
Available at: https://pypi.org/project/wordcloud/
[Accessed November 2022].

Python.org, 2022. *docs.python.org.* [Online]
Available at: https://docs.python.org/3/library/time.html
[Accessed November 2022].

Python, 2022. *docs.python.org.* [Online]
Available at: https://docs.python.org/3/library/os.html
[Accessed November 2022].

Rosenthal, M., 2022. *Must-Know Phishing statistics: Updated 2022.* [Online]
Available at: https://www.tessian.com/blog/phishing-statistics-2020/
[Accessed Aprik 2022].

Seaborn, 2022. *seaborn.pydata.org.* [Online]
Available at: https://seaborn.pydata.org/
[Accessed November 2022].
Security, I., 2022. *X-Force Threat Intelligence Index 2022,* s.l.: IBM Security.
Sharma, H., Meenakshi, E. & Bhatia , S. K., 2017. *A comparative analysis and awareness survey of phishing detection tools.* Bangalore, India, IEEE.
Stapf-Fine, H. &. B. U. &. B. S. &. D. T. &. E. R. &. R. M. &. S. F. &. S. A., 2018. *Policy Paper on the Asilomar Principles on Artificial Intelligence.*, s.l.: s.n.
Subasi, A., Molah, E., Almkallawi, F. & Chaud, T. J., 2017. *Intelligent phishing website detection using random forest classifier.* Ras Al Khaimah, United Arab Emirates, IEEE.
Union, E., 2018. *EU General Data Protection Regulation (GDPR),* s.l.: European Union.
Verizon, 2021. *Data breach investigations report.* [Online]
Available at: https://www.verizon.com/business/resources/reports/dbir/
[Accessed April 2022].
Yang, P., Zhao, G. & Zeng, P., 2019. *Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning.* s.l., IEEE.
Yang, W., Zuo , W. & Cui, B., 2019. *Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network.* s.l., IEEE.

9 VIDEO PRESENTATION

This is the link to my presentation, it is only accessible by People in the National College of Ireland with the Link

- https://studentncirl-my.sharepoint.com/:p:/g/personal/x20191359_student_ncirl_ie/ESVrmrZvZHROjVMGZ5k_E2sBAlq3tUA9P5iYIBdA8y8e9g?e=4mlhPB

10 ACKNOWLEDGMENT

Firstly, would also like to acknowledge Ross Spelman at National College of Ireland, my assigned supervisor during this research project. I am very thankful and grateful for all the help, supervision and little nudges given by him during the research. His valuable input and comments helped progress my research topic from an idea to a completed project. I would like to acknowledge and thank all the lecturers and fellow students involved during this Masters. I could not single out one as all have been excellent and a pleasure to work with. I would like to express my gratitude to all the authors who have contributed to the literature that has been used in this thesis. Their work provided me with the foundation for my own research.
I would like to acknowledge and thank Seán & Méabh for their support, love, and encouragement. Especially for putting up with me over the last two years. Without their unquestioned support this accomplishment would not have been possible. Finally, I would like to thank to all those who have supported me in the completion of this thesis. Your contributions have been invaluable, and I am truly grateful for all your help.
Thank you all, Mike.