Detection of Phishing Emails Through

Natural Language Processing and Supervised Machine Learning

In partial fulfillment of

the requirements for the Honors Program

By

Elijah Williams

Iona University, New Rochelle, NY

February-2023

©2023 by Elijah Williams

All Rights Reserved

**ABSTRACT**

People often communicate through emails because of its ease of access and the multiple uses that emails can provide. Emails can be utilized in various forms and because of the versatility of email communication, sensitive and confidential information can be distributed through these emails. Phishing is a method of obtaining this sensitive/confidential information from emails by tricking people into revealing this information. Additionally, phishing can be defined as a social engineering technique that is deployed to obtain sensitive/confidential information from victims by tricking the victim into believing that malicious host is a reputable host. Phishing attacks have characteristics that allow for easier distinguishing between safe emails and malicious emails since attackers cannot take the identity of the reputable source and they all have a certain goal when employing their attacks. For this research, the main approach of detecting these phishing emails was through extracting features (key words, parts of speech, etc.) from the emails and creating a dataset from those features. Afterwards, supervised machine learning algorithms (Decision Trees, K-Nearest Neighbors, etc.) would use the dataset to make predictions/classifications on whether an email was safe (ham) or phishing. Highest results for the algorithms currently mark at 93.50% with a false negative rate of 06.02%.

Keywords: Phishing, Machine Learning, Natural Language Processing, Decision Trees, Support Vector Machines, K-Nearest Neighbors

Table of Contents

[1. INTRODUCTION 5](#_Toc121143498)

[**1.1 Problem** 5](#_Toc121143499)

[**1.2 Objective** 5](#_Toc121143500)

[**1.3 Approach** 5](#_Toc121143501)

[**1.4 Organization** 6](#_Toc121143502)

[2. BACKGROUND 6](#_Toc121143503)

[**2.1 Key Concepts** 6](#_Toc121143504)

[**2.1.1 Phishing** 6](#_Toc121143505)

[**2.2 Literature Review** 7](#_Toc121143506)

[2.2.1 Phishing Email Detection Through Three Types of Data 8](#_Toc121143507)

[2.2.2 Phishing Website Detection 9](#_Toc121143508)

[2.2.3 Phishing Website Detection II 9](#_Toc121143509)

[3. APPROACH 10](#_Toc121143510)

[**3.1 High Level Design** 10](#_Toc121143511)

[**3.2 Features** 11](#_Toc121143512)

[**3.3 Classifiers** 13](#_Toc121143513)

[**3.4 Implementation** 16](#_Toc121143514)

[4. METHODOLOGY, RESULTS, AND ANALYSIS 21](#_Toc121143515)

[**4.1 Dataset** 21](#_Toc121143516)

[**4.2 Results** 22](#_Toc121143517)

[**4.3 Analysis** 24](#_Toc121143518)

[5. CONCLUSIONS 25](#_Toc121143519)

[**5.1 Summary** 25](#_Toc121143520)

[**5.2 Potential Impact** 25](#_Toc121143521)

[**5.3 Future Work** 26](#_Toc121143522)

[REFERENCES 27](#_Toc121143523)

# **1. INTRODUCTION**

## **1.1 Problem**

Phishing emails have been a very successful method of harvesting credentials from unsuspecting victims for a variety of reasons. Cyber criminals have used this type of cyber-attack in multiple forms that serve different purposes in reaching their desired goals. At our current state of defense against phishing email attacks, many methods rely on human nature to recognize the tendencies of these phishing emails and discern safe/dangerous emails by themselves. There have been attempts to use technology to identify phishing attacks, all with various methodologies and approaches that focus on aspects of phishing that other researchers differ from. Increasing defenses against phishing emails is of vital importance, due to the potential financial losses and security breaches for individuals and businesses alike.

## **1.2 Objective**

The goal of this research is to provide another method for the phishing email attack utilized by cyber criminals. By providing a technological means of defense, in addition to the human safety measures advocated by cybersecurity professionals, the effectiveness of phishing email attacks should decrease and decrease the utilization of these attacks.

## **1.3 Approach**

The classifications of these phishing emails will be performed by using supervised machine learning methods on a dataset that we have created from the raw emails collected. The dataset will consist of features that were extracted from the raw emails by the preprocessing methods used for our research. Features will be extracted from the emails using a Natural Language Processing library that will simplify the words into their base form and record the occurrence of key words within an email. Additionally, the parts of speech will be extracted from the emails to record the percentages of each part of speech within the email.

## **1.4 Organization**

Chapter 2 will cover key background information that will serve as the basis for the rest of the knowledge necessary to comprehend and interpret the research, including the works of other. researchers and explain how the methodology/approach of their research differs from the research that will be presented within this paper. Chapter 3 will cover the implementation design and approach of the program. This will cover which features were chosen and for what purpose, which classifiers were chosen and their purpose, and how the techniques that were explained at a high level were done in practice. Chapter 4 will cover the completed dataset and its properties, along with an overview of the results found by the classifiers and what that means for our research. Chapter 5 will consist of a conclusion with a summary of what the research we have performed means, how did we reach the goals that we did, and how the research can be further improved on.

# **2. BACKGROUND**

## **2.1 Key Concepts**

### **2.1.1 Phishing**

Phishing, in the simplest of terms, is the practice of sending in fraudulent messages under the guise of being a reputable source with the intention of tricking the recipients of the phishing messages into revealing personal information (credit card information, login information, etc.) to the attacker. By performing these types of attacks, attackers can yield access to various amounts of information that are beneficial to their purposes. Phishing, as an attack, is classified as a social engineering attack which utilizes psychological manipulation into tricking users into a security mistake. While phishing is a broad range of attacks, phishing can be classified into four common versions of the attack. The four common versions of the attack are:

1. Spear Phishing – This version of the phishing attack has a designated target as its victim and crafts the attack into having higher likelihood of success (Alkhalil). The spearing analogy originates from the fact that you can target a specific fish with a spear compared to a fishing pole.
2. Whaling – This version of the phishing attack is a subset of spear phishing by targeting a person who is high up in a company or in a position of power (Alkhalil). The whaling analogy arises from the fact that a whale is a target that is even bigger than a fish.
3. Smishing – This version of the phishing attack is distributed through text messaging or short message service (SMS) which has a clickable link within the message or phone number to call back to (Alkhalil).
4. Email Phishing – The most common form of the phishing attack, this version distributes the attack to any email addresses within their possession and try to harvest personal information from the victims through clickable links and social engineering tactics (Alkhalil). This type of phishing attack is the central focus of this research.

With all these options (and many more) as potential avenues for attacks, that underlies the reason why phishing attacks have increased in both number and volume. Research has found that about 90% of companies/organizations have been targeted by these types of attacks in some capacity. Since the people within these institutions are frequently targeted because they are high profile candidates to the attackers, which is why, along with human education and preparation for these attacks, a technical approach can be utilized as well in the protection of these phishing attacks (Alkhalil).

## **2.2 Literature Review**

Various other research studies within the field of phishing emails vary in the approach made to the analyzation of the phishing attacks. A popular approach found within this field of study is the research on phishing websites and how these websites differ from the typical legitimate email. This method of analysis does not place a significant level of importance on the efforts of social engineering and rather focuses on the information that can be revealed from the phishing websites domain and the analysis of responses and other key features. In a much smaller subsection of phishing analysis, there can be found analysis on phishing emails. The analysis on phishing emails varies from how the analysis performed in this study persists as the methodology used within that research focused on analysis technique mixing the analysis of URL and few key words that appeared within the email. The methodology that will be used within this research focuses more on the social engineering aspect of phishing attacks which checks multiple key words in addition to the types of word used to engineer these attacks.

### 2.2.1 Phishing Email Detection Through Three Types of Data

The research performed by Rawal, Rawal, Shaheen, and Malik focuses primarily on the detection of phishing emails through machine learning techniques and data processing. The methodology behind their approach to the detection of phishing emails lies in extracting key features from the emails which then leads to a dataset that machine learning techniques can be performed on (Rawal). The primary features that were extracted from the emails can be categorized into the following three categories:

1. Link-Based
2. Tag-Based
3. Word-Based

From these three displayed categories, a dataset consisting of nine fields was created and passed to the machine learning algorithms for analysis (Rawal). The research and methodology that is being proposed within this analysis differs by placing more emphasis on the word-based approach and creating a larger dataset. From there, it focuses on various words and the natural language of the emails to the end goal of producing a dataset that holds key words and the percentages of parts of speech in the emails.

### 2.2.2 Phishing Website Detection

The research performed by Mahajan and Siddavatam focuses on the detection of phishing websites by pulling out features that can be found within the website itself. Some of the features that can be detailed in the research are whether a URL is redirected and various other features that can be extracted from a URL (Mahajan). By performing this type of analysis on the phishing websites, the researchers observe the behavior of websites and determine what type of behavior and responses from websites illicit phishing qualities of websites or legitimate types of websites (Mahajan). The research, intended for this paper, handles phishing emails and the natural language that attackers employ to coerce victims into clicking on links within the email. This practice will allow for the analysis of phishing emails without the opening of links that occur when checking for certain attributes of URL features.

### 2.2.3 Phishing Website Detection II

The research performed by Krishna et al. also places a primary focus on phishing websites compared to phishing emails for the primary analysis. By placing a larger focus on the websites and the behavior of the website compared to the word choice used within the phishing email, their research aimed at the identification of phishing websites without the factor of social engineering in the attacks. As attackers develop more and more nuanced ways of deceiving people, an underlying factor beneath all the attacks will be convincing the user to land on the website and give their credentials. The research that was performed within this paper focuses on those manipulation tactics of the attackers in the phishing emails. By placing more emphasis on the word choice within the emails compared to the websites of the email, the tricks/trends that the attackers use become more apparent.

# **3. APPROACH**

## **3.1 High Level Design**

The preprocessing system of this methodology makes use of Python and the NLP library that exists in the python libraries (NLTK). Through use of multiple libraries in the Python language, a dataset that was suitable for supervised machine learning analysis was implemented. The utilized models for the analysis of the dataset used techniques, such as k-fold cross validation and other techniques geared toward their respective model, to perform analysis and infer which emails were phishing emails and ham emails. The methodology behind each level of the process in the proposed system is explored in more detail in the following sections of the **Proposed System**.

The creation of the dataset takes place in a multiple phase process. First, phishing emails and ham emails had to be collected from online resources for the analysis of the project. The phishing emails (.eml files) were collected from the phishing corpus and the ham emails were found in a resource on Kaggle that listed all the ham emails (.txt files). From there, the emails must be within the same format for proper analysis to be performed, therefore, the ham emails were converted into the same format that the phishing emails consisted of. The files are now prepared to be fed into a preprocessor and have various features extracted.

The preprocessor extracts various features from the emails but the first feature that is given to each of the emails is whether an email is a phishing email or a ham email. These labels are given to each email depending on which source (in the directory) the email came from and labels it accordingly. Then, using Python’s BeautifulSoup library and string manipulation, emails are cleaned of extra text that would influence the speed and efficiency of extracting features from the emails. Sequentially, using Python’s NLTK library, each email was broken up word-by-word and features were extracted according to those listed in **Features**.

Once every email has been analyzed, all the features that have been extracted are exported into a csv file that can be used for machine learning analysis. From there, machine learning models and results are explored using the classifiers found in **Classifiers** and results are detailed in those found in **Results**.

## **3.2 Features**

**Word-based Features** – Word-based features consist of checking the number of times a certain word appears in an email. Words that have the same meaning but are written in different forms were stemmed to reduce the dimensionality of the csv file for the dataset. For example, the word “run” and “running” were considered the same word since they both contain the stem word of “run.” An example of a few words that were checked for in the emails were the following:

|  |  |
| --- | --- |
| **Key Words** | **Reasoning** |
| Account | Phishers are trying to harvest the credentials of users, therefore, “account” appears often in a phishing email. |
| Alert | Phishers are aiming to rattle a person into performing actions without doing the proper inspections of a legitimate email, therefore, stating “alert” raises likelihood of error. |
| Confidential | Often, phishers make statements stating that confidential credentials have been exposed and a password reset is required which would lead to the stealing of the victim’s information |
| Fraudulent | A common approach used by attackers is to send an email reporting that there has been fraudulent activity performed by your account and credentials change is required. |
| Information | An attacker can bring up the word “information” in variety of ways when trying to trick a victim into revealing something personal to them. However, this word tends to get frequently utilized when attempting to force the victim into action. |
| Notification | Similar to “alert”, the attacker is attempting to rattle the user and raise the likelihood of error on their part. Therefore, the victim will receive a “notification” or “be notified” about account changes or actions among that nature. |
| Password | One of the most valuable pieces of information that an attacker can receive is a victim’s password because of the tendency of people to use the same password for multiple websites or accounts. Therefore, an attacker can state something along the lines of a password being changed and needing a reset in order to get the victim’s password. |
| Key Action Verbs | The purpose of phishing attacks is to get the victim to revealing their information while they still believe that the attacker is a trustworthy source. Therefore, action verbs that invoke the sense of urgency are often used within the emails and their attempts to trick the victim. |
| POS Features | Parts-of-Speech (POS) consist of the percentages of how much each part of speech appears in the email. The parts of speech that were checked for were the number of times the following appeared in the emails:   * Nouns * Verbs * Adjectives * Adverbs |

Table : Features of Dataset

## **3.3 Classifiers**

A classifier is an algorithm which implements the methods to classify the data given to it into classes. The classifiers utilized within the research are supervised classifiers, therefore, the classifiers will be trained with predefined classifications for the datapoints (emails). The following classifiers implemented in the research are explained below:

**Support Vector Machine** – Support Vector Machine is a supervised machine learning model that uses classification algorithms for two group classification problems. Support Vector Machines separate the datasets given to them by **hyperplanes**. Hyperplanes, essentially, are methods of dividing the information given to them by a dimension that is one smaller than the dimensions of the dataset. For example, if the given dataset can exist in a two-dimension plane, then the dataset will be separated by a one-dimensional hyperplane. In situations where a linear boundary could not be suited for the dataset, Support Vector Machines also make use of **kernels**. Kernels allow for the use of non-linear boundaries between the datasets when a linear boundary will not suffice. The below figures detail two situations where Support Vector Machines are utilized for the separation and classification of data.

Chart, scatter chart

Description automatically generated

Figure : SVM Linear Hyperplanes

Chart, scatter chart

Description automatically generated

Figure : SVM Polynomial Hyperplanes

Figure 1 depicts the support vector machine separating the dataset by a linear, one-dimensional hyperplane while the Figure 2 depicts the support vector machine separating the dataset by a non-linear, one-dimensional hyperplane.

**Decision Tree** – Decision Tree is a supervised machine learning model that will be utilized to categorize classes based on how previous questions were answered in the tree. A classification Decision Tree predicts that a dataset will be the most common occurrence at that prediction node. To build up a decision tree, there are various methods but a common practice for building up the decision tree is to use *entropy* to calculate a node’s “purity.” After doing this process, a series of questions/decisions will be created that classify the points in the dataset. The figure below illustrates a sample of a classification Decision Tree.

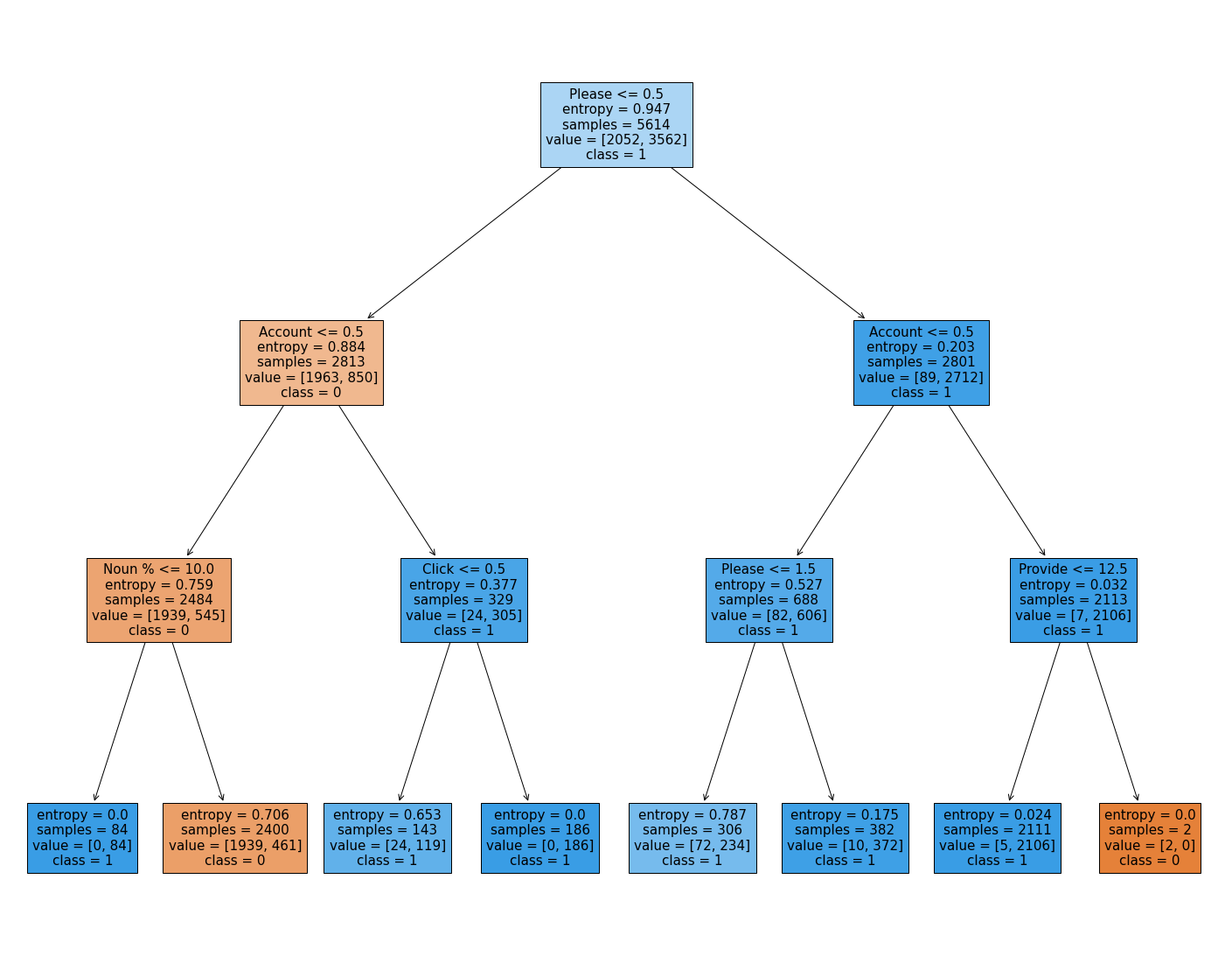


Figure : Decision Tree Sample Model

**K-Nearest Neighbors** – The K-Nearest-Neighbors model is a supervised machine learning model that categorize an unknown data point depending on the neighboring data points surrounding the unknown data point. Firstly, the K-Nearest Neighbors model finds a ***k*** number of nearest neighbors depending on some positive integer. After that, whichever class is more abundant amongst the neighbors will classify the unknown dataset. The figure below illustrates a K-Nearest Neighbors algorithm with a choice of three neighbors to classify the unknown data point.

A picture containing chart

Description automatically generated

Figure : K-Nearest Neighbors Sample Model

There are six dark datapoints and light datapoints, however, only the three closest datapoints to the unknown datapoint are considered for the classification portion of the model. As seen in the figure, since there are two dark neighbors and one light neighbor, the unknown datapoint will be classified as a light datapoint.

## **3.4 Implementation**

The conversion of the emails from their raw forms into features that can be learned upon within a dataset took place within multiple stages. Within the preprocessor, features were extracted from the emails in a variety of formats. However, for some features to be extracted, the emails had to be cleansed and manipulated to get an accurate representation of the features within the email. The following snippets of code will detail how certain features were extracted from the dataset, including the process of manipulating the emails to extract the features from the dataset.

Figure : perform\_pos\_tag(tokenized) function

The above figure describes the function utilized to extract the parts of speech percentages within each of the emails. The function takes a list of all the words as a parameter to the function with the expected output to be the percentage of nouns, verbs, adjectives, and adverbs within in the email. The function labels each word by a tag given to it by the “pos\_tag()” function and, by using that tag, the counter for that part of speech is increased. After every word has been properly analyzed, the calculation for the percentages is performed and ready to be appended to the dataset for each of the emails.



Figure : clean\_html(html) function



Figure : remove\_punctuations(words) fucntion



Figure : remove\_stopwords(words\_no\_punc) function

The above figures were implemented for the purpose of increasing efficiency with the creation of the dataset. Essentially, these functions were cleansing the emails for unnecessary information that could bring potential negative side effects into the dataset. Figure 6 cleanses the emails of potential html tags that could have remained within the email files. The purpose of this research was to notice trends and patterns solely through natural language; therefore, the html tags would not be able to provide useful information for the interpretation of our dataset. Figure 7 cleanses the emails of all punctuation marks that remained after converting the email from an entire string into a token of words. Punctuation marks do not provide much insight for what is important, and the central message being provided in the email, therefore, those punctuation marks are removed to reduce the noise that must be parsed through for the preprocessor. Figure 8 cleanses the emails of words that do not provide much insight into the email (i.e., the word “the” or “a”). Since these words do not have much value in the context of the research, this function removes this final step of words that limits the amount of unnecessary noise for our preprocessor.



Figure : stemming\_words(clean\_words) function

The above figure details how the testing for key words were created by the technique of “stemming.” Figure 9 takes the remaining words from the cleansing process and stems those words into their base form. In essence, this will allow words that are in different tenses to be counted as the same word, even though, they have the same meaning. For example, “ran” and “running” will be considered the same word in the context of stemming because they will both be considered as “run.” This function performs this task by making use of “PorterStemmer()” and “stems” the words according to its criteria.



Figure : perform\_statistics(stemmed) function

The above figure details how each of the keywords in the dictionary were accounted for. Figure 10 takes the list of all the stemmed words as a parameter to the function and outputs the number of occurrences of the key stems that were chosen for experimentation. The stemmed words chosen for experimentation are listed within the “word\_check” variable and increments the email statistics each time the stem appears within the email.

# **4. METHODOLOGY, RESULTS, AND ANALYSIS**

## **4.1 Dataset**

The analysis of the project focused on a dataset that consisted of both ham and phishing emails. There was a total of 7,017 emails analyzed within this dataset. Of those 7,017 emails, 4,466 (64%) of those emails were classified as phishing. The remaining 2,551 (36%) of those emails were classified as ham emails. From all these emails, there were a total of twenty features extracted from each of the emails.

## **4.2 Results**

A false positive, in the context of this research, describes an email that was a ham email but was classified as a phishing email. A false negative, in the context of this research, describes an email that was a phishing email but was classified as a ham email. Correctly classified emails will be predicted correctly based on the label given to them in the dataset.

The highest results of each classifier can be seen in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Classifier** | **Accuracy (%)** | **False Positive (%)** | **False Negative (%)** |
| Support Vector Machines | 90.5 | 3.38 | 12.45 |
| Decision Trees | 93.5 | 8.13 | 6.02 |
| K-Nearest Neighbors (Split) | 93.4 | 11.02 | 9.82 |
| K-Nearest Neighbors (Cross-Validation) | 92.0 | 12.19 | 8.82 |

Table : Highest Results from Classifiers

The result of each kernel for the support vector machine can be seen in the following table:

|  |  |
| --- | --- |
| **Kernels** | **Accuracy (%)** |
| Linear | 90.385 |
| Polynomial | 90.527 |
| RBF | 89.174 |
| Sigmoid | 71.368 |

Table : Support Vector Machine Classification Results

The result of every odd neighbor for the K-Nearest Neighbors (using the 80-20 split) can be seen in the following graph:

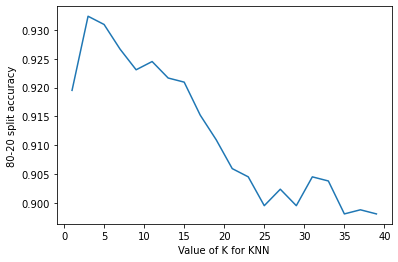
****

Figure : K-Nearest Neighbors (Split) Results

The result of every odd neighbor for the K-Nearest Neighbors (using the 10-fold cross validation) in the following graph:

Chart, line chart

Description automatically generated

Figure : K-Nearest Neighbors (Cross-Validation) Results

The result of deepening depths within the Decision Trees in the following graphs can be seen below:

Chart, line chart

Description automatically generated

Figure : Decision Tree Results

## **4.3 Analysis**

The results of the classifiers indicate the accuracy in which they were able to predict whether an email was a ham email or a phishing email. Looking beyond just the accuracy measures, however, the rates of the false positives and false negatives indicate how many false alarms were raised by the classifiers and how many dangerous/phishing emails were able to go undetected by the classifiers. Regarding this research, false negatives are more dangerous than a false positive because a false positive when implemented in the practical sense. Within the practical sense, a false positive implies that a safe email was marked/flagged as phishing which is not a huge concern. However, a false negative implies that a dangerous/phishing email managed to not be flagged and the recipient of the email could be misguided into believing the phishing email that he/she received is a safe email. Maintaining high accuracy, along with reducing the number of false negatives, is the best-case scenario for the classifiers and the classifier that handled this process the best was Decision Tree. The other classifiers managed to have high accuracies but failed to keep the percentage of false negatives as low as the Decision Tree classifier.

# **5. CONCLUSIONS**

## **5.1 Summary**

The emails used within the research were collected from online sources that contained both phishing emails and ham emails. By analyzing the word choice and manipulation within the emails, deciphering the methods that phishing attackers use to deceive victims becomes more apparent for the classifiers. The techniques that were used to decipher these attack patterns required a preprocessor that extracted key words from the emails and the percentages for certain parts of speech within the emails. By extracting these key features, supervised machine learning models were trained to detect emails that were phishing emails and emails that were ham(safe) emails. After implementing various supervised machine learning models on the dataset that was generated, the highest accuracies and least number of false negatives that were generated from the classifiers ended up at 93.50% accuracy and 06.02% false negative rate.

## **5.2 Potential Impact**

Phishing emails have always been a lucrative way for attackers to garner valuable information from victims. By having another layer of defense that can be used to detect phishing emails, potential victims of the attacks can be alerted to potential phishing attacks within their email provider. However, users still must be diligent in recognizing whether the emails that are in their inbox are indeed safe emails before interacting with them. Flagging emails that have the potential to be phishing emails provide users another indication for thoroughly checking an email of its validity. With this extra layer of defense for detecting phishing emails, less victims of phishing attacks should be feasible, and the effectiveness of prior phishing techniques should decrease in tandem.

## **5.3 Future Work**

The research detects phishing emails with a high accuracy and a low false negative rate; however, other studies have shown the potential other directions that can be used to detect phishing emails and websites. The research performed within this paper highlights text analysis for the detection of phishing emails to understand the methods that attackers have used to manipulate victims into being deceived by the emails. Emails, however, have aspects besides text such as: tags, website domains, etc., that can be analyzed to provide more insight on the technical creation of the phishing attacks. Implementing these aspects of emails into the dataset could garner higher accuracy rates and allow for greater defense of protecting victims from phishing attacks within their inboxes. With the addition of new features of the data, however, other concerns arise regarding the speech in which the classifier will train on and execute on when used in practical situations.

# **REFERENCES**

Alkhalil, Zainab, et al. “Phishing Attacks: A Recent Comprehensive Study and a New Anatomy.” *Frontiers*, Frontiers, 9 Mar. 2021, https://www.frontiersin.org/articles/10.3389/fcomp.2021.563060/full.

Krishna V, A., A, A., Jose, B., Anilkumar, K., & Lee, O. T. (2021, August 2). *Phishing detection using machine learning based URL Analysis: A Survey*. International Journal of Engineering Research & Technology. Retrieved September 29, 2022, from https://www.ijert.org/phishing-detection-using-machine-learning-based-url-analysis-a-survey

Mahajan, R., & Siddavatam, I. (2018, October). *Phishing website detection using machine learning algorithms - IJCA*. Phishing Website Detection using Machine Learning Algorithms. Retrieved September 29, 2022, from https://www.ijcaonline.org/archives/volume181/number23/mahajan-2018-ijca-918026.pdf

Rawal, Srishti, et al. “Phishing Detection in e-Mails Using Machine Learning.” *Research Gate*, Oct. 2017, https://www.researchgate.net/publication/320257918\_Phishing\_Detection\_in\_E-mails\_using\_Machine\_Learning.

Phishing Corpus (Dataset): <https://academictorrents.com/details/a77cda9a9d89a60dbdfbe581adf6e2df9197995a>.

Ham Emails (Dataset): https://www.kaggle.com/code/eswar3793/spam-filtering-hamnspam-data/data