Unveiling Phishing Patterns through Machine Learning

Hui Xu V01029720 Ke Xu V01037793 Renhou Zhang V01037648 Yihan Jiao V01045917

Guoyu Zhang V01047027 Yi Lu V00901547

***Abstract*—Phishing is one of the most prevalent and risky forms of cyberattacks nowadays. Attackers use phishing emails not only to disclose recipients' private information but also to install malware on recipients' devices. These actions bring risks and damage the security of personal information. We are going to preprocess the dataset from the UVic security department. In this report, we employ TF-IDF to extract features, to find words that have contributed to phishing emails. Since the data only contains phishing emails, supervised algorithms may not be suitable, we explored unsupervised techniques to obtain an insight into the patterns, features and characteristics of phishing emails. In addition, we also include the plan for further analysis in the next semester.**

***Index Terms*—phishing, cyberattack,TF-IDF, K-means**

I. Introduction

With the increasing number of cyberattacks, the security of personal information and devices becomes a critical issue. Phishing is one of the most risky and prevalent techniques. It can disclose sensitive information, hacking into digital devices and even remote control devices.

Machine Learning techniques are widely used in the field of cyber security. We use the dataset from the UVic security department to find the patterns, features, and characteristics. Given the dataset contains only text data and only phishing emails, it is not suitable to use supervised learning algorithms such as KNN, or SVM. In this report, the K-means algorithm is employed to cluster the data, since it is an unsupervised learning algorithm.

II. Problem Description

Our objective is to discern the underlying patterns, features, and characteristics within our dataset, which can be divided into three key components:

1) The initial challenge involves preparing the textual data for a machine learning algorithm.
2) Given the absence of ground truth labels, traditional supervised learning approaches may not be useful. Hence, determining a suitable alternative becomes paramount.
3) Once analyses are conducted, it's important to effectively visualize both the outcomes and the dataset itself to get insights efficiently.

III. Feature Extraction with TF-IDF

To understand which words are "important", we need to identify and highlight the key terms within a corpus of phishing emails. To achieve this, we employed the Term Frequency – Inverse Document Frequency (TF-IDF) technique. This is a widely used method in generating features in machine learning, thus helping to pinpoint the most relevant terms in phishing emails. TF-IDF accounts for both the term's prevalence within individual documents and its rarity across the entire corpus.

The subsequent sections will delineate the methodology adopted for TF-IDF vectorization, the resulting significant terms identified, and the insights gleaned from the analysis, which inform the development of an effective phishing email detection system. Here is a breakdown of the steps followed in our analysis:

1) **Data preparation:** We first concatenated the text from the 'Subject' and 'Body' fields of each email into a single column, ensuring any missing values were replaced with empty strings to maintain consistency in data processing.
2) **TF-IDF Vectorization:** We then utilized the 'TfidfVectorizer' from Scikit-learn, setting a limit of 100 features and excluding common English stop words to focus on the most informative terms. This vectorization transforms the text into a numerical format, with each term weighted according to its significance across the dataset.
3) **Feature Extraction:** The TF-IDF matrix was computed from the combined text, allowing us to quantitatively analyze the text data. Each row in the matrix corresponds to an email, while each column represents one of the terms identified by the vectorizer as a significant feature.
4) **Data aggregation:** We calculated the sum of TF-IDF scores for each term across all documents to determine their overall significance in the dataset.

The analysis yielded a list of the top 40 significant words, ranked by their cumulative TF-scores. These scores reflect the relative importance and uniqueness within the phishing email corpus.

As shown in the Figure 1, we noticed that certain words like "uvic", "ca", "https", and "com" receive high scores, suggesting that the imbalance in the dataset itself: many email samples have included words like "uvic", "ca", "https", and "com". This makes sense as the corpus consists predominantly of sample emails from the UVic domain, terms associated with this domain should appear significant.

uvic	293.437704
ca	272.992810
email	262.529029
https	220.166416
com	218.794120
message	211.579095
sent	194.906111
information	185.394136
university	179.604706
victoria	177.870058
notice	160.954700
account	156.320005
links	153.378619
outside	153.087126
sensitive	152.661300
cautious	152.386162
2022	139.577050
hello	123.698703
mail	119.470423
helpdesk	108.679714
dear	101.246458
payment	96.734101
regards	96.162918
update	95.364364
reply	93.418932
contact	93.362995
details	91.382439
click	90.981473
www	87.845384
password	85.561057
address	84.401900
thank	78.985431
mailbox	76.439344
link	76.229285
service	70.187655
thanks	70.079074
http	69.195165
time	69.062564
good	68.595172
new	68.231064
dtype: float64	

Fig. 1. **Top 40 frequency Words.**

This also means our preprocessing is not good enough, as the list of stop words used by 'TfidfVectorizer' does not include domain-specific terms or parts of URLs by default. In our case, these URLs should also be considered as common words and should be eliminated in the preprocessing process.

For improvements on preprocessing, we are trying to use the following methods:

1) Extend the stop words list to include common but uninformative words specific to this dataset. Adding words like "https," "com," "ca," and "uvic" could be beneficial to generate better results from TF-IDF.
2) URL Parsing: Extract and remove URLs or treat them specially, isolating domains or paths that might be of interest while discarding generic components.
3) Feature Engineering: For the URLs with spam purposes, try to clean the URLs with their actual hyperlink.
4) Detailed Separation of Dataset: Instead of considering 'Subject' and 'Body' as a whole in one sample, try to separate them into two independent groups, and do feature extraction from each of the groups to see if there are any interesting insights.

IV. Results and Visualization

A. Word Cloud Visualization

We used the TF-IDF results to generate a word cloud. Each word in the word cloud represents a significant word in the result of TF-IDF to be a possible feature. The larger the word, the more significant the importance when identifying phishing.

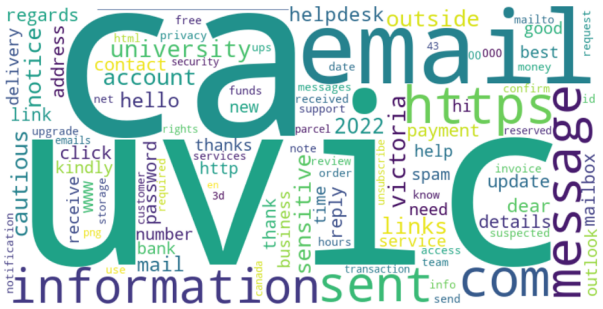

Fig. 2. **Word Cloud.**

B. Heat Map Visualization

We create a heat map to visualize the first 10 email documents from the dataset, emphasizing the most significant keywords based on their TF-IDF scores. Keywords like "account," "payment," and "transaction" appear on the x-axis, while the y-axis shows the indices of these documents. The color intensity in the heat map reflects the TF-IDF scores, with warmer colors representing a higher importance of the keyword in an email, and cooler colors showing lower importance.

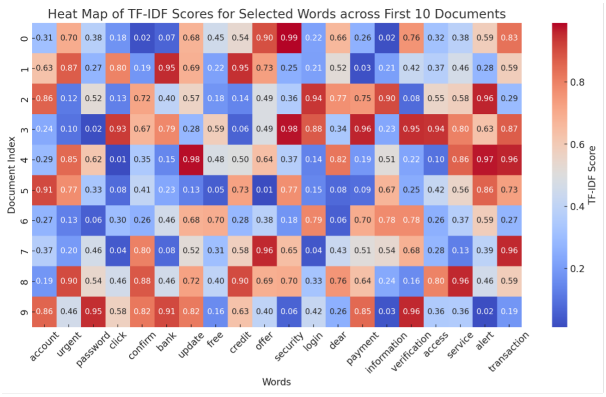

Fig. 3. **Heat Map Visualization.**

As shown in Figure 3, keywords such as "payment" and "transaction" have many red cells among the 10 email documents, indicating their relative significance in these phishing emails, matching our expectation.

C. K-means Clustering

We use the K-means clustering algorithm to analyze the structural similarities among phishing emails. We create a matrix where each email's "Subject" and "Body" texts' are combined into rows, and the columns represent the TF-IDF scores for each term found in the emails. We restrict the TF-IDF features to a maximum of 500 per text component (i.e., "Subject" and "Body"), totaling up to 1000 features for each email. We then apply K-means clustering to group these features into five distinct clusters. To facilitate the visualization of our clustering, we conduct Principal Component Analysis (PCA) to compress the data to two dimensions, enhancing the clarity of the visual representation on a 2D plot.

As illustrated in Figure 4, each email is depicted by a cross, with colors denoting its cluster membership. The cluster centers are marked by five red stars. Although these are initial results, they reveal some interesting clusters that are worth future investigation. Additionally, two of the clusters show some overlap in the 2D display, indicating the potential for improved dimensionality reduction techniques in subsequent analyses.

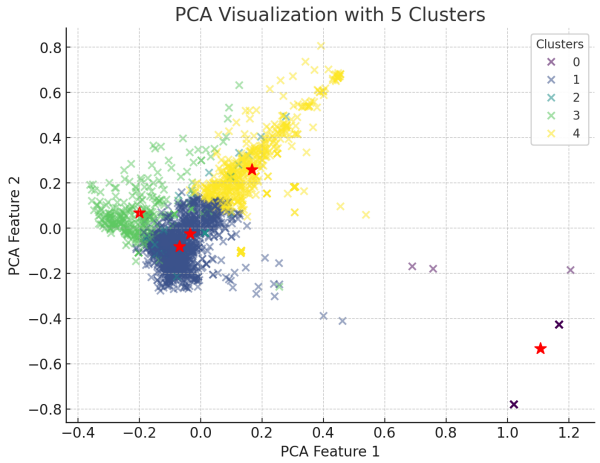

Fig. 4. **K-means Clustering Visualization.**

V. Conclusion

Based on the K-means results, phishing emails can be clustered into three clusters, which indicates a further analysis of the relationships between these clusters and what common features of emails in the same cluster are worth.

Another finding is high-frequency words may not necessarily contribute to the phishing emails, such as uvic or ca. These words almost appear in every email in the dataset, and they come from the warnings of UVic. A proper way is to filter the warnings and only deal with the phishing content.

Further analysis of URLs in the email is also practical and useful, since phishing URLs may contain multi-domain names, improper format, and obfuscation.

References

[1] Salloum, Said, Tarek Gaber, Sunil Vadera, and Khaled Shaalan. "A systematic literature review on phishing email detection using natural language processing techniques." IEEE Access 10 (2022): 65703-65727.

[2] Yaseen, Qussai. "Spam email detection using deep learning techniques." Procedia Computer Science 184 (2021): 853-858.

[3] Bountakas, Panagiotis, Konstantinos Koutroumpouchos, and Christos Xenakis. "A comparison of natural language processing and machine learning methods for phishing email detection." In Proceedings of the 16th International Conference on Availability, Reliability and Security, pp. 1-12. 2021.

[4] Ahmed, Naeem, Rashid Amin, Hamza Aldabbas, Deepika Koundal, Bader Alouffi, and Tariq Shah. "Machine learning techniques for spam detection in email and IoT platforms: analysis and research challenges." Security and Communication Networks 2022 (2022): 1-19.