Replace with Your Project Title

Abstract

Phishing is an evident cybersecurity threat. Cybercriminals often pose as reputable organizations and send emails with links to phishing websites to unsuspecting individuals. Individuals who enter phishing websites, expose themselves to data breaches and malware. Companies spend time and money investing in cybersecurity training to protect their employees and customers from phishing attempts. Nevertheless, phishing is still prevalent in today's society. In an October 2022 study, conducted by security provider SlashNext, found more than 255 million phishing attempts in email, mobile, and browser channels. SlashNext reports that there has been a 61

Introduction

The section includes SEVERAL paragraphs summarizing your project. It is like the extended version of Abstract - you may use one paragraph for each of these parts - problem statement, dataset description, machine learning algorithms you will use to solve the problem, experimental results, and how your solutions are better as compared to existing solutions. Please try to limit Introduction to one page.

Related Work

This section summarizes existing solutions to the problem or similar problems. Please try to categorize these existing techniques and provide some discussion on the pros and cons of them. Don't forget to include references to any existing work you mention.

Our Solution

This section elaborates your solution to the problem.

Description of Dataset

The first step in building the phishing detection model website is to choose an appropriate dataset that will consist of both phishing and legitimate websites, which will be used for training a model for predicting phishing URLs. The chosen dataset can be found in Kaggle, and is called “Web page Phishing Detection Dataset”, by Shashwat Tiwari. The dataset consists of 11430 rows of records containing URLs, and 87 columns that make up the features. This is a good dataset based on the fact that there are a large number of records and features to allow for a good breakdown of training and testing data. This will help us avoid either bias or variance while training the model.

This machine learning model will place an emphasis on the following aspect of the URLs: Punctuation and Word Choice. In order to achieve this, all of the records will be used. The features that will be extracted to train the model are the following: "https_token", "ratio_digits_url", "nb_hyphens", "nb_dots", "nb_underscore", "nb_slash". Some of these features are discrete (0 or 1) and some are continuous values. Features that hold discrete values help describe whether something is present or not in the URL (i.e hyphens), while continuous values quantify a feature of the URL (i.e ratio of digits). The values "phishing" and "legitimate" will be encoded to 0 and 1, respectively.

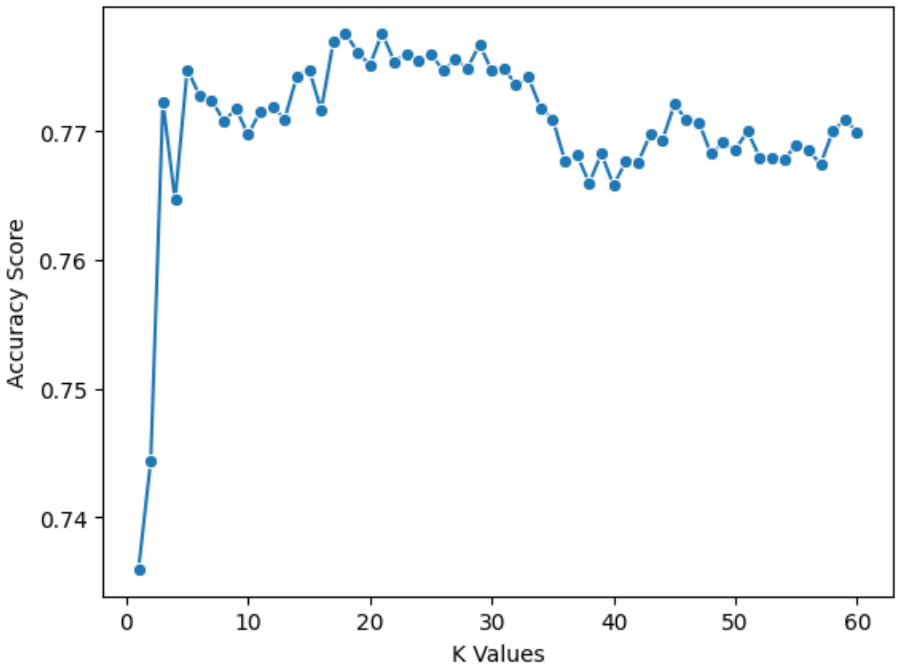
	https_token	ratio_digits_url	nb_hyphens	nb_dots	nb_underscore	nb_slash	status
0	1	0.000000	0	3	0	3	legitimate
1	1	0.220779	0	1	0	5	phishing
2	0	0.150794	1	4	2	5	phishing
3	1	0.000000	0	2	0	2	legitimate
4	1	0.000000	2	2	0	5	legitimate

Machine Learning Algorithms

One machine learning we will be using is KNN-classifier.

Implementation Details

The date is split into a training size of 75 percent, and a dataset for testing of size 25 percent of the entire dataset. To optimize this algorithm, we are collecting the best value of k ’best_k’, which is shown in the graph below:



Comparison

This section includes the following: 1) comparing the performance of different machine learning algorithms that you used, and 2) comparing the performance of your algorithms with existing solutions if any. Please provide insights to reason about why this algorithm is better/worse than another one.

Future Directions

This section lays out some potential directions for further improving the performance. You can image what you may do if you were given extra 3-6 months.

Conclusion

This section summarizes this project, i.e., by the extensive experiments and analysis, do you think the problem is solved well? which algorithm(s) might be better suitable for this problem? Which technique(s) may help further improve the performance?
Last but not the least, don't forget to include references to any work you mentioned in the report.