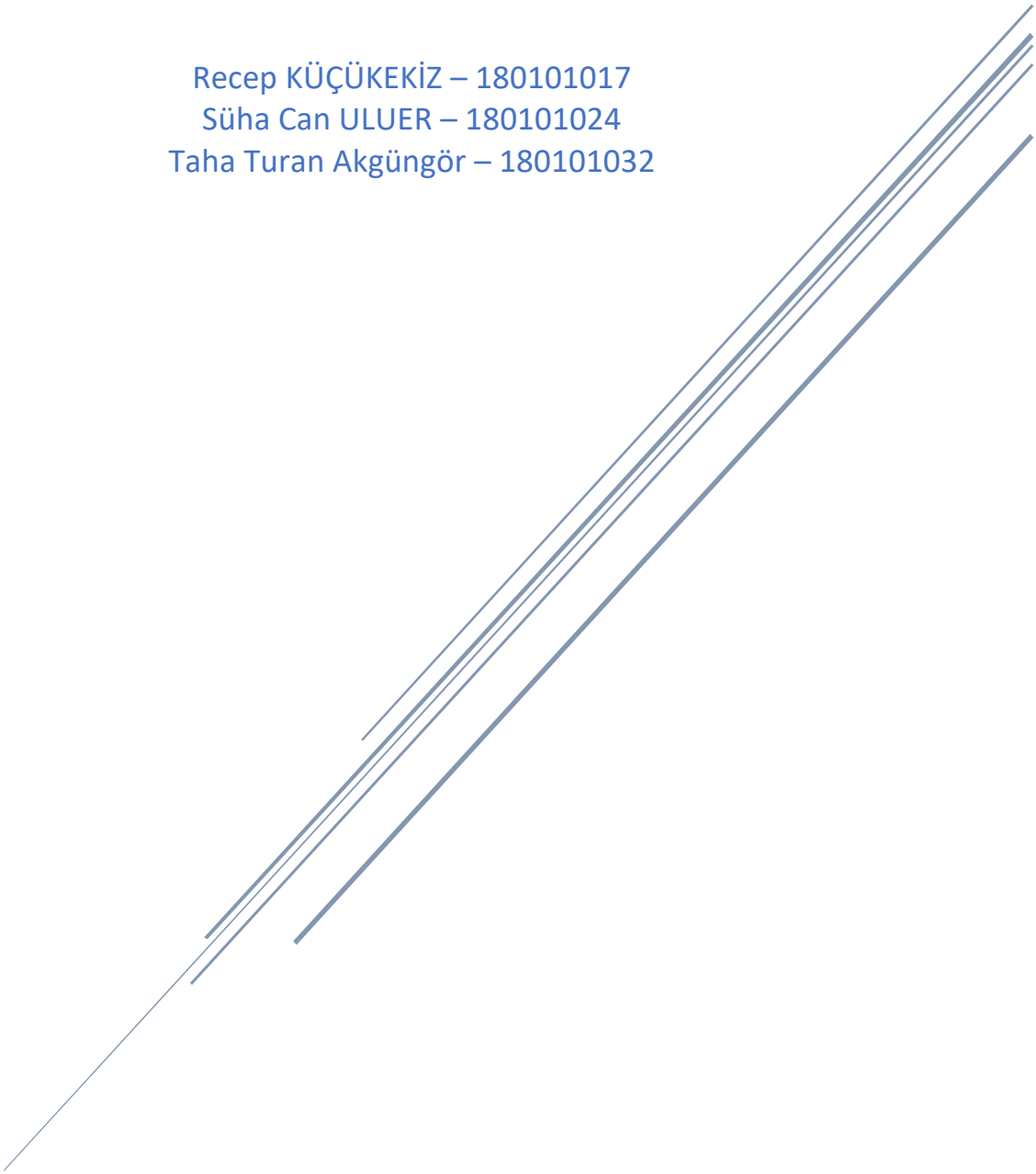
CYBER SECURITY

Detecting Phishing Website with Machine Learning

Recep KÜÇÜKEKİZ – 180101017
Süha Can ULUER – 180101024
Taha Turan Akgüngör – 180101032



Adana Alparslan Türkeş Science and Technology University
Computer Engineering

CONTENTS

1. INTRODUCTION

Phishing is the most common social engineering and cyber attack. In this type of attack, a phisher targets unsuspecting online users and tricks them into disclosing confidential information to be used fraudulently. To avoid phishing, users should be aware of phishing websites, have a blacklist of phishing websites that requires information about the website detected as phishing, detect them when they appear early using machine learning and deep neural network algorithms. The method based on machine learning of the above three methods, has proven to be more effective than other methods. Despite this, online users continue to fall into the trap of disclosing sensitive information on phishing websites.

2. OBJECTIVE

A phishing website is a common social engineering method that impersonates trusted uniform resource locators (URLs) and web pages. The goal of this project is to train machine learning models and deep neural networks on the generated dataset to predict phishing websites. Both phishing and innocuous URLs of websites are collected to create a dataset, from which the required URL and website content-based features are extracted. The performance level of each model is measured and compared.

3. APPROACH

Below are the steps involved in completing this project:

- Collect the dataset containing phishing and legitimate websites from open source platforms.
- Write a code to extract the required features from the URL database.
- Analyze and preprocess the dataset using preprocess techniques.
- Split the dataset into training and test sets.
- Run selected machine learning and deep neural network algorithms such as SVM, Random Forest, Decision Tree, XGBoost, Multilayer Perceptron (Deep Learning) on the dataset.
- Write a code to display the evaluation result taking into account accuracy metrics.
- Compare the results obtained for the trained models and determine which one is better.

4. DATA COLLECTION

Legitimate URLs were randomly collected from a source on GitHub, numbering 3500. Phishing URLs were collected from an open source service called PhishTank. This service provides a set of phishing URLs in various formats such as csv, json, etc. That are updated hourly. The 3500 URLs from this collection were randomly selected.

5. FEATURE EXTRACTION

In this step, features are extracted from the URLs dataset. The extracted features are categorized into:

- Address Bar based Features
- Domain based Features
- HTML & Javascript based Features

5.1. Address Bar Based Features

Many features can be extracted that can be considered as address bar based features. The following features were evaluated for this project.

- IP Address in URL
  - Checks if the URL has an IP address. URLs can have an IP address instead of a domain name. If the URL uses an IP address as an alternative to the domain name, we can be sure that someone is trying to steal personal information with this URL.
  - If the domain part of the URL has an IP address, the value assigned to this property is 1 (phishing) or 0 (legal).
- "@" Symbol in URL
  - Checks for an '@' symbol in the URL. The use of the '@' symbol in the URL causes the browser to ignore everything before the '@' symbol, and the actual address usually follows the '@' symbol.
  - If the '@' symbol is present in the URL, the value assigned to this property is 1 (phishing) or 0 (legal).
- Length of URL
  - Calculates the length of the URL. Phishers can use long URL to hide the suspicious part in the address bar. In this project, if the length of the URL is greater than or equal to 54 characters, then the URL classified as phishing is otherwise legitimate.
  - If the length of the URL is >= 54, the value assigned to this property is 1 (phishing) or 0 (legitimate).
- Depth of URL
  - Calculates the depth of the URL. This property calculates the number of subpages in the given url based on '/'.
  - The value of the property is a numeric value based on the URL.
- Redirection "//" in URL
  - Checks for the presence of "//" in the URL. The presence of "//" in the URL path means that the user will be redirected to another website. The position of "//" in the URL is calculated. We found that if the URL starts with "HTTP", it means that "//" should appear in the sixth position. However, if "HTTPS" is used in the URL, "//" should appear in the seventh position.
  - If "//" is anywhere in the URL other than after the protocol, the value assigned to this property is 1 (phishing) or 0 (legitimate).
- "http/https" in Domain name
  - Checks for "http/https" in the domain part of the URL. Phishers can add the "HTTPS" token to the domain name of a URL to trick users.

    - If "http/https" is present in the domain part of the URL, the value assigned to this property is 1 (phishing) or 0 (legal).
- Using URL Shortening Services “TinyURL”
    - URL shortening is a method on the "World Wide Web" whereby the length of a URL can be significantly reduced and still redirect to the required web page. This is accomplished through an "HTTP Redirect" on a short domain name that connects to the web page with a long URL.
    - If using URL Shortening Services, the value assigned to this property is 1 (phishing) or 0 (legal).
- Prefix or Suffix "-" in Domain
    - Checking for a '-' in the domain part of the URL. The hyphen symbol is rarely used in legitimate URLs. Phishers tend to add prefixes or suffixes separated by (-) to the domain name to make users feel that they are dealing with a legitimate web page.
    - If the '-' symbol is present in the domain part of the URL, the value assigned to this property is 1 (phishing) or 0 (legitimate).

5.2. Domain Based Features

Many features can be extracted that can be considered as address bar based features. The following features were evaluated for this project.

- DNS Record
    - For phishing websites, the alleged identity is not recognized by the WHOIS database or no record for the hostname is found. If the DNS record is empty or cannot be found, the value assigned to this property is 1 (phishing) or 0 (legal).
- Website Traffic
    - This feature measures the popularity of the website by determining the number of visitors and the number of pages they visit. However, phishing websites may not be recognized by the Alexa database because they live for a short period of time (Alexa the Web Information Company., 1996). Analyzing our dataset, we see that in the worst-case scenarios, legitimate websites are among the top 100,000. Also, if the domain has no traffic or is not recognized by the Alexa database, it is classified as "Phishing".
    - If the domain's rank is < 100000, the value of this attribute is 1 (phishing) and 0 (legitimate).
- Age of Domain
    - This feature can be removed from the WHOIS database. Most phishing websites live for a short period of time. The minimum age of a legitimate domain name is considered to be 12 months for this project. The age here is nothing but the difference between creation and expiration time.
    - If the domain age is > 12 months, the value of this property is 1 (phishing) and 0 (legitimate).
- End Period of Domain
    - This property can be extracted from the WHOIS database. The remaining domain time for this property is calculated by finding the difference between the expiration time and the current time. For this project, the expiration time considered for a legitimate domain is 6 months or less.

- If the domain expiration time is > 6 months, the value of this attribute is 1 (legitimate) or 0 (legal).

5.3. HTML and JavaScript Based Features

Many features can be extracted that can be considered as address bar based features. The following features were evaluated for this project.

- IFrame Redirection
  - IFrame is an HTML tag used to display an additional web page inside a currently displayed web page. Phishers can use the "iframe" tag and make it invisible, i.e. without frame borders. In this context, phishers use the "frameBorder" property, which causes the browser to create a visual identification.
  - If the inner frame is empty or no response is found, the value assigned to this property is 1 (phishing) or 0 (legal).
- Status Bar Customization
  - Phishers can use JavaScript to show users a fake URL in the status bar. To extract this property, we must extract the web page source code, specifically the "onMouseOver" event, and check if it makes any changes to the status bar.
  - If the response is empty or found on mouseover, the value assigned to this property will be 1 (phishing) or 0 (legitimate).
- Disabling Right Click
  - Phishers use JavaScript to disable the right-click functionality so that users cannot view and save the web page source code. This feature is handled exactly as "Using onMouseOver to hide the link". For this feature though, we will look for the "event.button==2" event in the web page source code and check if right-click is disabled.
  - If the response is empty or onmouseover is not found, the value assigned to this property will be 1 (phishing) or 0 (legal).
- Website Forwarding
  - The fine line that separates phishing websites from legitimate ones is the number of times a website is redirected. In our dataset, we see that legitimate websites were redirected at most once. On the other hand, phishing websites that include this feature were redirected at least 4 times.

6. MACHINE LEARNING MODELS

This is a supervised machine learning task. There are two main types of supervised machine learning problems, called classification and regression. Since the input URL is classified as phishing (1) or legal (0), this dataset falls under the classification problem. The machine learning models (classification) considered to train the dataset in this study are:

- Decision Tree
- Random Forest
- Multi-Layer Perceptron (Deep Learning)
- XGBoost
- Support Vector Machine

6.1. Descision Tree

A decision tree is a type of supervised machine learning used to categorize or make predictions based on how previous questions were answered. The model is a form of supervised learning, meaning the model is trained and tested on a set of data containing the desired classification.

The decision tree may not always provide a clear answer or decision. Instead, it can present options so that the data scientist can make an informed decision on their own. Decision trees mimic human thinking, so it is often easy for data scientists to understand and interpret the results.

6.2. Random Forest

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

6.3. Multi-Layer Perceptron

The field of artificial neural networks is often just called neural networks or multi-layer perceptron's after perhaps the most useful type of neural network. A perceptron is a single neuron model that was a precursor to larger neural networks.

It is a field that investigates how simple models of biological brains can be used to solve difficult computational tasks like the predictive modeling tasks we see in machine learning. The goal is not to create realistic models of the brain but instead to develop robust algorithms and data structures that we can use to model difficult problems.

The predictive capability of neural networks comes from the hierarchical or multi-layered structure of the networks. The data structure can pick out (learn to represent) features at different scales or resolutions and combine them into higher-order features, for example, from lines to collections of lines to shapes.

6.4. XGBoost

XGBoost, also known as Extreme Gradient Boosting, is a supervised learning technique that uses an ensemble approach based on the Gradient boosting algorithm. It is a scalable end-to-end tree boosting system that is widely used by data scientists to achieve state-of-the-art results on many machine learning challenges. It can solve both classification and regression problems and achieve better results with minimal effort.

The first version of this algorithm was implemented using Gradient Boosting machines. Later, after making this work open source, a large community of data scientists started contributing to XGBoost projects and further improved this algorithm. With the help of such a great community, XGBoost has become a software library and can be installed directly on our systems. It supports various interfaces such as Python, R, C++, Julia and Java.

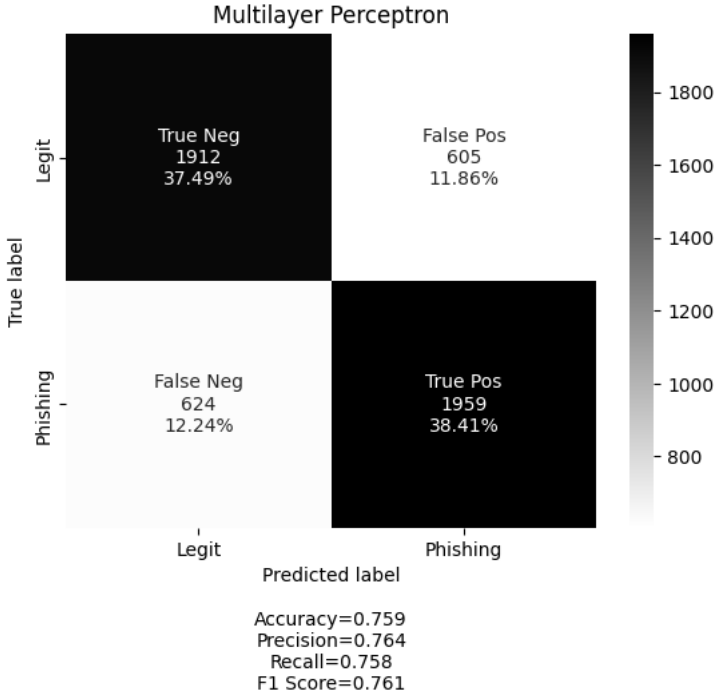
6.5. Support Vector Machine

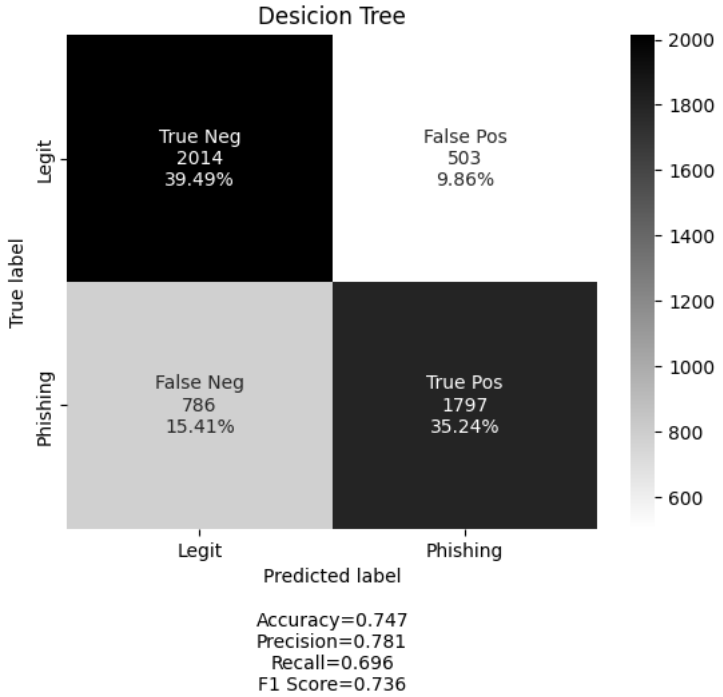
Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

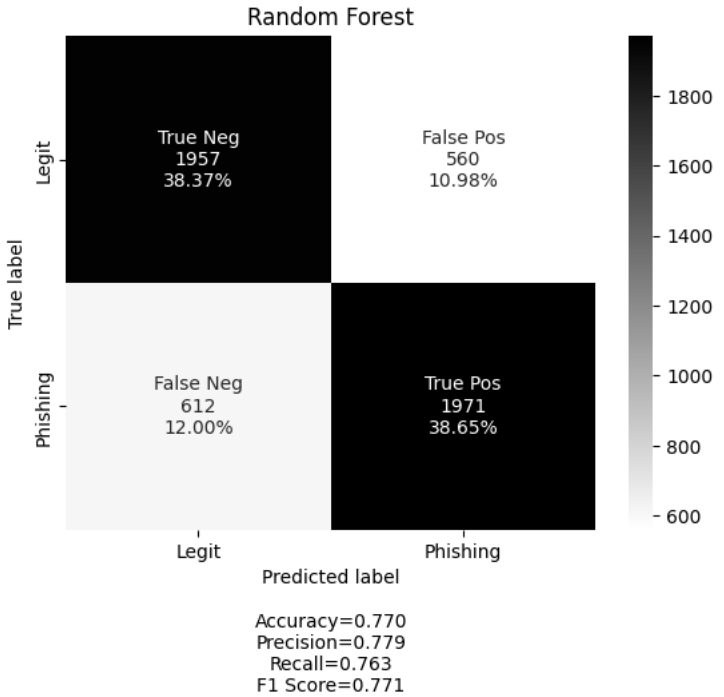
The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

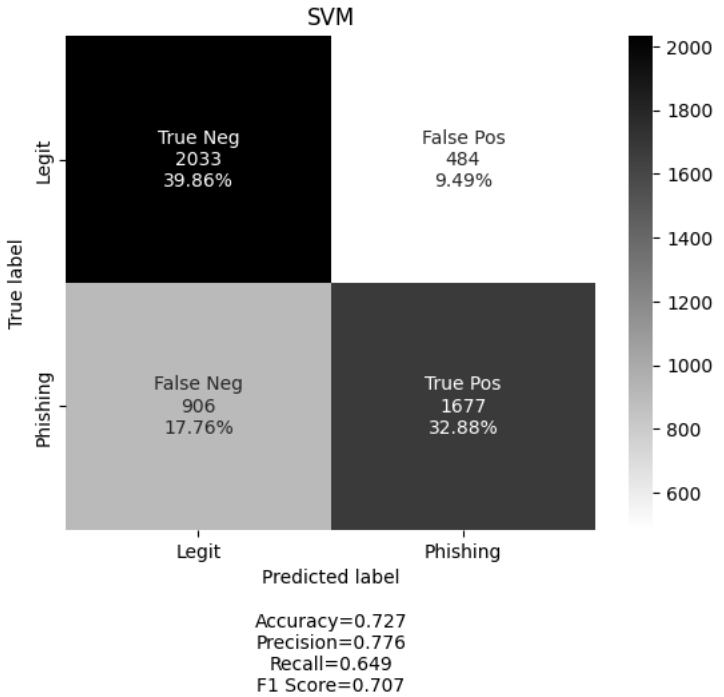
7. MODEL EVALUATION

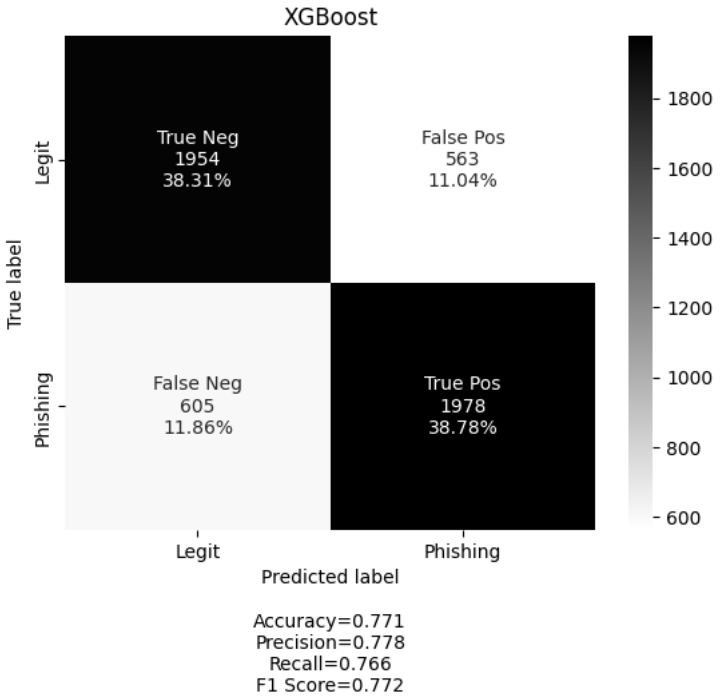
SVM, Random Forest, Decision Tree, XGBoost, Multilayer Perceptron (Deep Learning) algorithms are applied on 17k dataset. Accuracy, Precision, Recall, F1 Score and Confusion Matrix are shown below for each algorithm train result.











ML Model	Test Accuracy
XGBoost	77.1%
Random Forest	77.02%
Multilayer Perceptron	75.9%
Desicion Tree	74.73%
SVM	72.75%

Multilayer Perceptron is a Deep Learning based algorithm. It run 427 iterations with the dataset. As a result of these metrics, XGBoost algorithm has best accuracy score. Because of this we extract the model and designed a simple website for testing. You can see screenshot of the website on below.

8. TESTING ON WEBSITE

Website shows is the given URL safe or unsafe.