VIETNAM GENERAL CONFEDERATION OF LABOUR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**Phạm Ngọc Linh - 521H0360**

**Lâm Nguyễn Anh Thy - 521H0377**

**Nguyễn Đình Việt Hoàng - 522H0120**

**DETECT PHISHING WEBSITES USING MACHINE LEARNING**

**FINAL PROJECT**

**INTRODUCTION TO INFORMATION SECURITY**

**HO CHI MINH CITY, YEAR 2024**

VIETNAM GENERAL CONFEDERATION OF LABOUR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**Phạm Ngọc Linh - 521H0360**

**Lâm Nguyễn Anh Thy - 521H0377**

**Nguyễn Đình Việt Hoàng - 522H0120**

**DETECT PHISHING WEBSITES USING MACHINE LEARNING**

**FINAL PROJECT**

**INTRODUCTION TO INFORMATION SECURITY**

**Instructor**

**Miss. Huỳnh Ngọc Tú**

**HO CHI MINH CITY, YEAR 2024**

**ACKNOWLEDGEMENT**

We sincerely thank Miss. Huỳnh Ngọc Tú for teaching us the Introduction to Information Security course with great enthusiasm. We want to express our deep appreciation for the dedication and professional knowledge that you shared with us. Through your classes, we gained a better understanding of the fundamental aspects of the Introduction to Information Security, thanks to your detailed explanations and practical applications. You helped us grasp the knowledge and apply it effectively. Finally, we extend our heartfelt gratitude to Miss. Huỳnh Ngọc Tú for your commitment and invaluable support throughout our learning journey in this course. The skills and knowledge we acquired will continue to impact our future development. We sincerely thank you and wish your health, success, and happiness.

*Ho Chi Minh City, May 10, 2024*

*Authors*:

*Phạm Ngọc Linh*

*Lâm Nguyễn Anh Thy*

*Nguyễn Đình Việt Hoàng*

**THE COMPLETION REPORT HAS BEEN SUBMITTED AT TON DUC THANG UNIVERSITY**

Our group assures that this is our own report and was guided by Miss. Huỳnh Ngọc Tú. The research content and results in this report are honest and have not been published in any form before. The figures in the tables used for analysis, comments, and evaluations were collected by the authors from various sources clearly stated in the reference section.

Additionally, the report includes some comments, evaluations, and data from other authors and organizations, all of which are cited and noted for their origin.

**If any fraud is detected, we fully take responsibility for the content of our final report for the second semester of the 2023-2024 academic year.** Ton Duc Thang University is not involved in any copyright or intellectual property violations that we may cause during the process (if any).

*Ho Chi Minh City, May 10, 2024*

*Authors*:

*Phạm Ngọc Linh*

*Lâm Nguyễn Anh Thy*

*Nguyễn Đình Việt Hoàng*

**DETECT PHISHING WEBSITES USING MACHINE LEARNING**

**ABSTRACT**

Phishing is an online threat where an attacker impersonates an authentic and trustworthy organization to obtain sensitive information from a victim. One example of such is trolling, which has long been considered a problem.One of the most common forms of phishing is through URLs, where attackers disguise malicious URLs as legitimate ones to trick users into clicking on them. Machine learning techniques have shown promise in detecting phishing URLs, but their effectiveness can vary depending on the approach used.

However, recent advances in phishing detection, such as machine learning-based methods, have assisted in combatting these attacks. Therefore, this paper develops and compares three models for investigating the efficiency of using machine learning to detect phishing domains. It also compares the most accurate model of the three with existing solutions in the literature. These models were developed using artificial neural networks (ANNs), support vector machines (SVMs), decision trees (DTs), and random forest (RF) techniques. In this report, our group focus on three models that are decision trees (DTs), and random forest (RF) techniques and support vector machines (SVMs).

TABLE OF CONTENTS

[LIST OF FIGURES vi](#_Toc153688675)

[ABBREVIATIONS vii](#_Toc153688676)

[CHAPTER 1. INTRODUCTION AND OVERVIEW OF THE TOPIC 1](#_Toc153688677)

[1.1 The reason for choosing the topic 1](#_Toc153688678)

[1.2 Research Methodology 1](#_Toc153688679)

[1.2.1 In theoretical terms 1](#_Toc153688680)

[1.2.2 In terms of programming 2](#_Toc153688681)

[1.2.3 In terms of operation 2](#_Toc153688682)

[1.3 Scope of the Project 4](#_Toc153688683)

[CHAPTER 2. THEORETICAL FOUNDATIONS 5](#_Toc153688684)

[2.1 Introduction to the website 5](#_Toc153688685)

[1.2 Programming Languages 5](#_Toc153688686)

[2.2.1 Node.JS 5](#_Toc153688687)

[CHATER 3. ANALYSIS OF SYSTEM DESIGN 11](#_Toc153688688)

[3.1 Problem Statement 11](#_Toc153688689)

[3.2 System specification 12](#_Toc153688690)

[3.2.1 Functions of Admin 12](#_Toc153688691)

[3.2.2 Employee functions 13](#_Toc153688692)

[3.3 Analysis and design 13](#_Toc153688693)

[3.3.1 Specification 13](#_Toc153688694)

[3.3.2 Relationship model 15](#_Toc153688695)

[CHAPTER 4. EXPERIMENT 22](#_Toc153688696)

[4.1 Design 22](#_Toc153688697)

[4.2 Account management 22](#_Toc153688698)

[4.3 Product management 24](#_Toc153688699)

[4.3 Employee management 26](#_Toc153688700)

[4.4 Customer management 27](#_Toc153688701)

[4.5 Report management 30](#_Toc153688702)

# LIST OF FIGURES

[Image 3.1 Entity Relationship Diagram 15](#_Toc153688648)

[Image 3.2 Activity diagram login 16](#_Toc153688649)

[Image 3.3 Activity diagram View Product Details 16](#_Toc153688650)

[Image 3.4 Activity diagram View Customer Invoice Details 17](#_Toc153688651)

[Image 3.5 Activity diagram View Employee Details 17](#_Toc153688652)

[Image 3.6 Activity diagram Payment for Orders 18](#_Toc153688653)

[Image 3.7 Activity diagram Generate Report 19](#_Toc153688654)

[Image 3.8 Activity diagram Add Product 20](#_Toc153688655)

[Image 3.9 Activity diagram Edit Product 20](#_Toc153688656)

[Image 3.10 Activity diagram Add or Edit Employee 21](#_Toc153688657)

[Image 4.1 Login screen 22](#_Toc153688658)

[Image 4.5 Category screen 24](#_Toc153688662)

[Image 4.6 Product details screen 25](#_Toc153688663)

[Image 4.7 Add and edit products screen 25](#_Toc153688664)

[Image 4.8 List of employee screen 26](#_Toc153688665)

[Image 4.9 Staff details screen 26](#_Toc153688666)

[Image 4.10 Add and edit employees screen 27](#_Toc153688667)

[Image 4.11 List of customers screen 27](#_Toc153688668)

[Image 4.12 List of orders screen 28](#_Toc153688669)

[Image 4.13 Invoice details screen 28](#_Toc153688670)

[Image 4.14 Transaction processing screen 29](#_Toc153688671)

[Image 4.15 Pay screen 30](#_Toc153688672)

[Image 4.16 Report list screen 30](#_Toc153688673)

[Image 4.17 Detailed report screen 31](#_Toc153688674)

# ABBREVIATIONS

|  |  |
| --- | --- |
| URL | Uniform Resource Locator |
| SMS | Short Message Service |
| DTA | HyperText Markup Language |
| RFA | Cascading Style Sheets |
| SVM | User Experience |

# CHAPTER 1. INTRODUCTION AND OVERVIEW OF THE TOPIC

* 1. **What is phishing?**

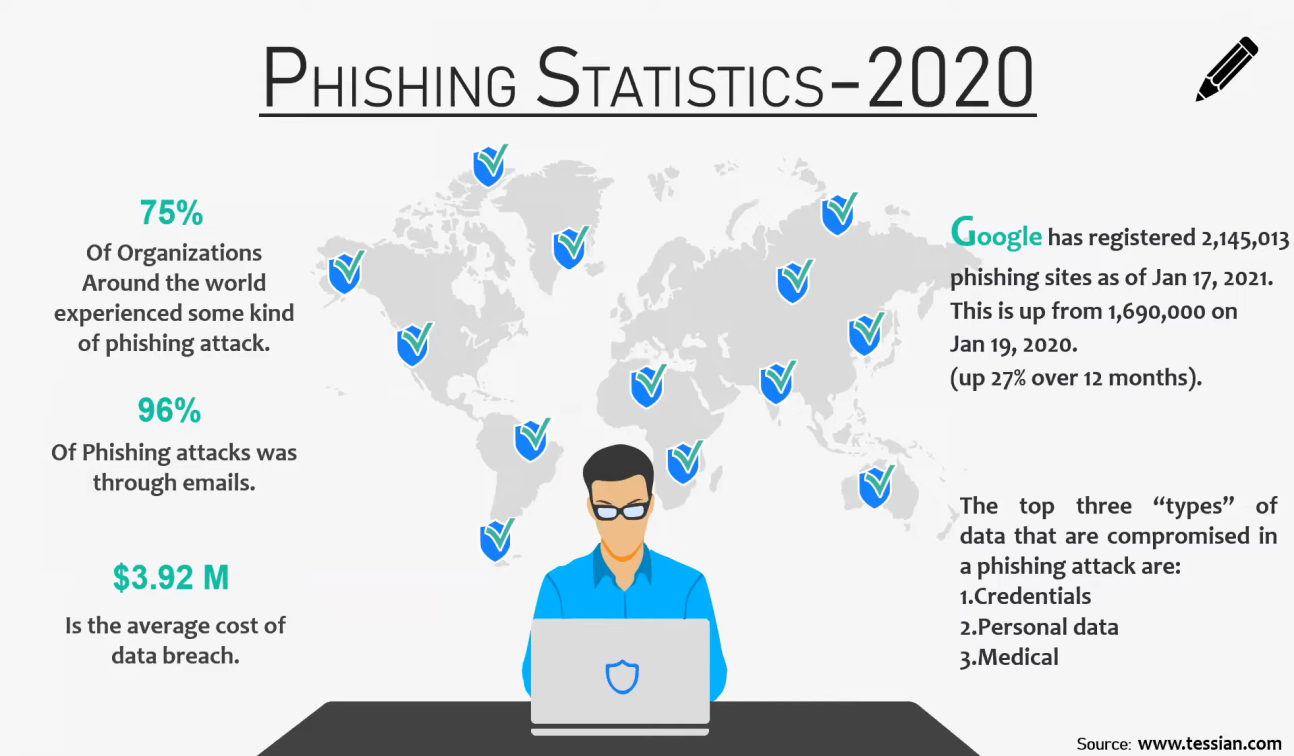


Image 1

* Phishing constitutes a cybercrime wherein individuals are contacted via email, telephone, or text message by imposters posing as legitimate entities, aiming to entice them into divulging sensitive information like personal identifiers, financial details, and passwords.
* Cybercriminals deploy various techniques, including the installation of malicious software on computers, to pilfer credentials, often resorting to intercepting usernames and passwords of users' online accounts. Phishers utilize diverse channels such as email, URLs, instant messages, forums, telephone calls, and texts to extract user information.
* The structure of phishing content mirrors original content, deceiving users into accessing it with the intent to acquire their sensitive data.
* The primary goal of phishing is to obtain personal information for financial gain or identity theft, resulting in substantial economic losses globally. Financial/payment institutions and webmail services are frequently targeted in phishing attacks, as highlighted by recent studies conducted by the Anti-Phishing Working Group (APWG).
* The inception of phishing lawsuits dates back to 2004 when a Californian teenager fabricated a replica of the "America Online" website, exploiting it to extract sensitive information and access credit card details for illicit monetary transactions. Apart from email and website phishing, cybercriminals employ techniques like 'vishing' (voice phishing), 'smishing' (SMS phishing), and various other evolving methods.
* Modern phishing attacks are sophisticated and increasingly challenging to discern. According to an Intel survey, 97% of security professionals struggle to distinguish between genuine emails and phishing attempts.
  1. **Classification of phishing attack techniques**
* Phishing websites are challenging to an organization and individual due to its similarities with the legitimate websites .There are forms of phishing attacks such as: (Các trang web lừa đảo gây khó khăn cho tổ chức và cá nhân do sự giống nhau với các trang web hợp pháp. Có các hình thức tấn công lừa đảo như sau:)
  + Technical subterfuge refers to the attacks include Keylogging, DNS poisoning, and Malwares. (Kỹ thuật lừa đảo liên quan đến các cuộc tấn công bao gồm Keylogging, DNS poisoning, và Malwares.)
  + In these attacks, attacker intends to gain the access through a tool / technique. On the one hand, users believe the network and on the other hand, the network is compromised by the attackers.
  + Social engineering attacks include Spear phishing, Whaling, SMS, Vishing, and mobile applications. (Các cuộc tấn công kỹ thuật xã hội bao gồm Spear phishing, Whaling, SMS, Vishing, và ứng dụng di động.)
  + In these attacks, attackers focus on the group of people or an organization and trick them to use the phishing URL.
* Apart from these attacks, many new attacks are emerging exponentially as the technology evolves constantly.

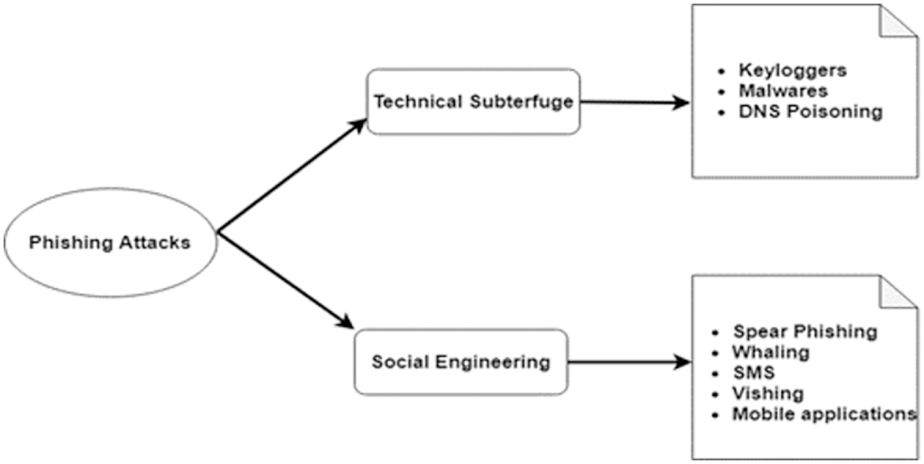


Image 1.2

* 1. **Phising detection approaches**

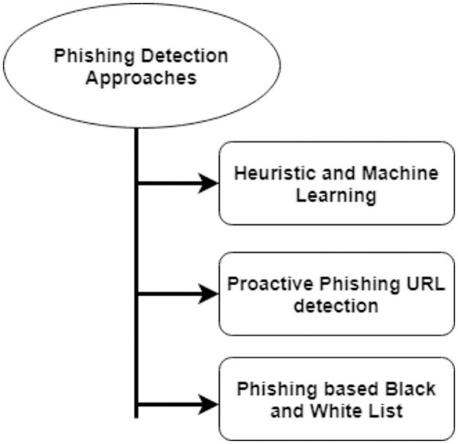


Image 1.3

* Heuristic and Machine Learning(Tiếp cận Hueristic và Học Máy):
  + Heuristic Approaches: These methods rely on predetermined rules or patterns to identify phishing attempts. This may involve analyzing various elements of emails or websites, such as URL structure, content, sender details, and visual indicators. (**Các Phương pháp tiếp cận Heuristic**: Những phương pháp này dựa vào các quy tắc hoặc mô hình đã được xác định trước để nhận diện các nỗ lực lừa đảo qua mạng của kẻ tấn công. Điều này có thể bao gồm việc phân tích các yếu tố khác nhau của email hoặc trang web, như cấu trúc URL, nội dung, chi tiết người gửi và các chỉ báo hình ảnh.) While heuristic approaches are easy to implement, they may struggle with detecting sophisticated attacks.
  + Machine Learning Approaches: Machine learning techniques can automatically learn patterns and attributes of phishing attacks from data. Features like email headers, URL properties, content, and sender details can be extracted and used to train machine learning models. (Các Phương pháp Học Máy: Các kỹ thuật học máy có thể tự động học các mẫu và thuộc tính của các cuộc tấn công phishing từ dữ liệu. Các đặc trưng như tiêu đề email, thuộc tính URL, nội dung, và chi tiết người gửi có thể được trích xuất và sử dụng để huấn luyện các mô hình học máy.) Models such as decision trees, random forests, support vector machines (SVM), or deep learning architectures can be utilized for classification. Machine learning approaches can adapt to evolving phishing tactics but necessitate labeled data for training.
* Proactive Phishing URL Detection: Proactive phishing URL detection entails identifying and blocking phishing URLs before they can reach potential victims. This typically involves continuously scanning the web for new phishing URLs and analyzing their characteristics to ascertain their malicious intent. (Phát hiện URL giả mạo một cách chủ động: Phát hiện URL giả mạo một cách chủ động liên quan đến việc xác định và chặn các URL giả mạo trước khi chúng có thể tiếp cận nạn nhân tiềm năng. Điều này thường liên quan đến việc quét liên tục trên web để tìm các URL giả mạo mới và phân tích các đặc điểm của chúng để xác định ý định độc hại của chúng.)
* Methods such as web crawling, content analysis, and comparing similarities with known phishing URLs can be employed for proactive detection. Upon identification, phishing URLs can be added to blacklists or flagged for further examination.
* Phishing-based Black and White Lists:
  + Blacklists: These comprise known phishing URLs, domains, IP addresses, or sender email addresses. Continuously updated based on the latest phishing threats, blacklists are used to prohibit access to malicious resources. When an incoming email or website request matches an entry on the blacklist, it is identified as potential phishing and can be blocked or subjected to additional scrutiny.
  + Whitelists: These contain trusted URLs, domains, IP addresses, or sender email addresses known to be legitimate. Whitelisting aids in reducing false positives by permitting access solely to recognized safe resources. However, maintaining whitelists can be challenging due to the dynamic nature of the web, and it may not be suitable for all environments.
* Danh sách các trang web đen và trắng:
  + Danh sách đen: Bao gồm các URL, tên miền, địa chỉ IP hoặc địa chỉ email người gửi đã biết là giả mạo. Được cập nhật liên tục dựa trên các mối đe dọa giả mạo mới nhất, danh sách đen được sử dụng để cấm truy cập vào các nguồn tài nguyên độc hại. Khi một email đến hoặc yêu cầu trang web khớp với một mục trong danh sách đen, nó được xác định là giả mạo tiềm năng và có thể bị chặn hoặc chịu sự kiểm tra thêm.
  + Danh sách trắng: Chứa các URL, tên miền, địa chỉ IP hoặc địa chỉ email người gửi được biết là hợp pháp. Danh sách trắng giúp giảm thiểu các kết quả dương tính giả mạo bằng cách chỉ cho phép truy cập vào các nguồn tài nguyên an toàn đã được công nhận. Tuy nhiên, việc duy trì danh sách trắng có thể gặp khó khăn do tính chất động của web, và nó có thể không phù hợp với tất cả các môi trường.
  1. **Steps to implement machine learning algorithms:**

Data collection:

* Collect data from legitimate websites and scam websites.
* Data can include website content, URLs, JavaScript code, etc

Data preprocessing:

* Eliminate noisy, incomplete data.
* Normalize data, such as converting URLs to a uniform form.
* Extract features from data.

Machine Learning model selection and training:

* Choose the right algorithm, e.g. Random Forest, Support Vector Machine, Convolutional Neural Network (CNN), etc.
* Divide the data into training set and test set.
* Train the model on the training set.

Model Rating:

* Evaluate the performance of the model on the test set.
* Adjust the model if necessary.

Deploy the model:

* Use trained models to detect new phishing websites.
* Update the model regularly to ensure detection efficiency.
  1. **Introduction to Machine Learning Algorithms Suitable for Phishing Detection**
* Machine learning algorithms offer a promising approach to phishing detection by leveraging patterns and features extracted from data to classify phishing websites accurately. Here are some commonly used machine learning algorithms suitable for phishing detection:
* Decision Trees:
  + Decision trees are intuitive models that recursively split the dataset based on features to make classification decisions.
  + They are easy to interpret and understand, making them suitable for generating rules for identifying phishing websites based on various features.
  + Decision trees can capture complex decision boundaries, allowing them to effectively distinguish between legitimate and phishing websites.
* Random Forests:
  + Random forests are an ensemble learning technique that combines multiple decision trees to improve classification accuracy and robustness.
  + They work by training multiple decision trees on random subsets of the dataset and aggregating their predictions through voting or averaging.
    - Random forests are less prone to overfitting compared to individual decision trees and can handle high-dimensional feature spaces effectively.
* Support Vector Machines (SVM):
  + SVM is a powerful supervised learning algorithm used for classification tasks.
  + It works by finding the optimal hyperplane that separates data points into different classes while maximizing the margin between classes.
  + SVM can handle nonlinear decision boundaries through the use of kernel functions, making it suitable for capturing complex relationships in phishing detection datasets.
* Neural Networks:
  + Neural networks, especially deep learning architectures, have shown promising results in various classification tasks, including phishing detection.
  + They consist of multiple layers of interconnected neurons that learn hierarchical representations of data.
  + Neural networks can automatically extract features from raw data and capture complex patterns, making them highly effective for phishing detection when trained on large datasets.
* Advantages of Machine Learning
  + Automation: Streamlines the process of identifying phishing attacks, saving time and resources.
  + Efficiency: Capable of accurately detecting new fraudulent websites.
  + Adaptability: Can learn and adjust to emerging phishing attack techniques.
* Drawbacks of Machine Learning
  + Data: Effective model training requires a significant amount of high-quality data.
  + Explanation: Challenges arise in elucidating the rationale behind the model's decisions.
  + Reliability: Regular updates to both data and the model are essential for sustained effective detection.
* Examples of Tools:
  + Google Safe Browsing: A service safeguarding users from malicious websites and downloads. It operates by warning users when attempting to access unsafe sites and assisting website owners by notifying them of potential compromises.
  + PhishTank: Enables users to report phishing websites and maintains an updated list of suspicious sites. Users can submit sites they suspect to be phishing attempts, and others can validate these submissions through voting, ultimately flagging potential phishing sites accordingly.

**CHAPTER 2: MACHINE LEARNING ALGORITHMS**

**2.1. DECISION TREE ALGORITHM**

***2.1.1. Decision tree algorithm overview***

* The decision tree algorithm is one of the most commonly utilized methods in machine learning technology. It is known for its simplicity in both understanding and implementation.
* The algorithm starts by selecting the best attribute for classification as the root of the tree. It then proceeds to build the tree until it reaches the leaf node. Each internal node in the tree represents an attribute, while each leaf node corresponds to a class label.
* Decision tree algorithms employ techniques such as the Gini index and information gain to calculate these nodes. The resulting training model can be used to predict target values or classes in a tree-like structure.

***2.1.2. Decision Tree Terminologies***

Before learning more about decision trees let’s get familiar with some of the terminologies:

* *Root Node:* The initial node at the beginning of a decision tree, where the entire population or dataset starts dividing based on various features or conditions.
* *Decision Nodes:* Nodes resulting from the splitting of root nodes are known as decision nodes. These nodes represent intermediate decisions or conditions within the tree.
* *Leaf Nodes:* Nodes where further splitting is not possible, often indicating the final classification or outcome. Leaf nodes are also referred to as terminal nodes.
* *Sub-Tree:* Similar to a subsection of a graph being called a sub-graph, a sub-section of a decision tree is referred to as a sub-tree. It represents a specific portion of the decision tree.
* *Pruning*: The process of removing or cutting down specific nodes in a decision tree to prevent overfitting and simplify the model.
* *Branch / Sub-Tree:* A subsection of the entire decision tree is referred to as a branch or sub-tree. It represents a specific path of decisions and outcomes within the tree.
* *Parent and Child Node:* In a decision tree, a node that is divided into sub-nodes is known as a parent node, and the sub-nodes emerging from it are referred to as child nodes. The parent node represents a decision or condition, while the child nodes represent the potential outcomes or further decisions based on that condition.

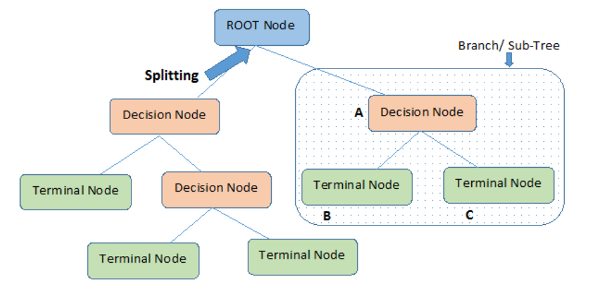


Image 2.1

Let’s understand decision trees with the help of an example:

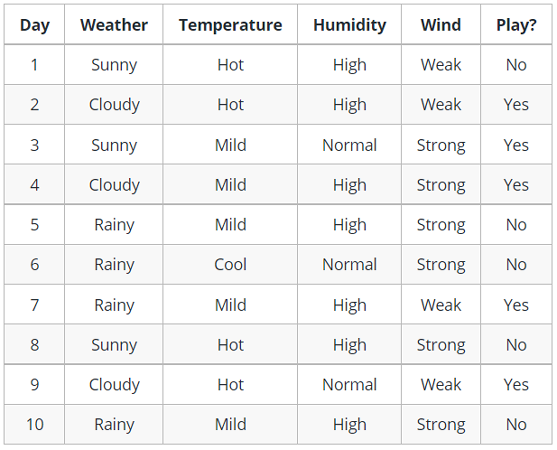


Image 2.2

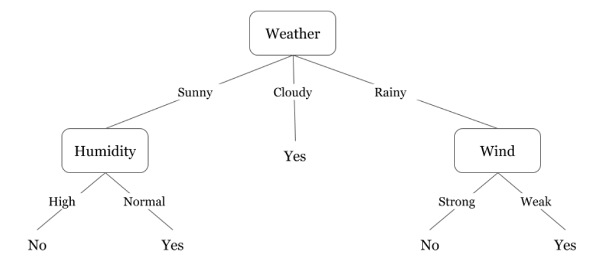


Image 2.3

* Decision trees are structured upside down, with the root positioned at the top and branching out into various nodes. Essentially, they can be likened to a series of if-else statements. At each step, the tree checks a condition and proceeds accordingly to the next attached node based on whether the condition is true or false.
* In the provided diagram, the tree first assesses the weather conditions—whether it's sunny, cloudy, or rainy. If it's rainy, it then considers factors like humidity and wind strength. For instance, if there's weak wind during rain, it suggests the possibility of playing outdoors.
* You might have noticed that if the weather is cloudy, the decision tree immediately concludes that it's suitable for playing. But why does it stop there without further branching? This is because, in the training dataset, the output for cloudy weather always results in a positive outcome for playing. Hence, there's no need to delve deeper into splitting the node.
* The objective of machine learning is to minimize uncertainty or disorder in the dataset, and decision trees help achieve this goal.
* Now, you might wonder how to determine the root node, decision nodes, and when to stop splitting. To make these decisions, we employ a metric called "Entropy," which measures the level of uncertainty in the dataset.

***2.1.3. Here's how decision trees classify phishing websites based on feature values:***

1. Root Node Selection:

- Initially, the decision tree considers all features available in the dataset.

- It selects the feature that best splits the dataset into subsets with the most distinct classes (phishing or legitimate websites).

- The selection criterion could be based on information gain, Gini impurity, or another measure of the purity of the resulting subsets.

2. Splitting Nodes:

- Once the feature is selected at the root node, the dataset is partitioned into subsets based on the values of that feature.

- Each subset represents a distinct branch or path in the decision tree.

- For example, if the selected feature is "URL length," the dataset might be split into two subsets: one containing URLs shorter than a certain threshold and another containing longer URLs.

3. Decision Rules:

- At each splitting node, a decision rule is formed based on the values of the selected feature.

- This decision rule dictates which subset a data point should follow based on the feature's value.

- For instance, if the decision rule at a node is "if URL length <= X, go to left child node; otherwise, go to right child node," it means that URLs shorter than or equal to X will follow the left branch, while longer URLs will follow the right branch.

4. Recursive Splitting:

- The process of selecting features and splitting the dataset is repeated recursively for each subset or branch of the decision tree.

- At each step, the algorithm selects the feature that maximizes the purity of the resulting subsets, aiming to separate instances of different classes as much as possible.

5. Leaf Node Formation:

- The process continues until a stopping criterion is met, such as reaching a maximum tree depth or having a minimum number of samples at each leaf node.

- Once the stopping criterion is reached, terminal nodes or leaf nodes are formed.

- Each leaf node represents a class label (phishing or legitimate), and the majority class of instances in that leaf node determines its classification label.

6. Classification of New Instances:

- When a new instance (website) is presented to the decision tree for classification, it traverses the tree from the root node down to a leaf node.

- At each node, the decision tree evaluates the feature values of the instance and follows the appropriate branch based on the decision rules.

- Once the traversal reaches a leaf node, the classification label associated with that leaf node is assigned to the instance, determining whether it is classified as phishing or legitimate.

* Decision trees classify phishing websites by recursively partitioning the dataset based on feature values, forming decision rules at each node to guide the traversal of the tree. This process continues until leaf nodes are formed, where classification decisions are made based on the majority class of instances in each leaf node.

**2.2. RANDOM FOREST ALGORITHM**

***2.2.1. Random forest algorithm overview***

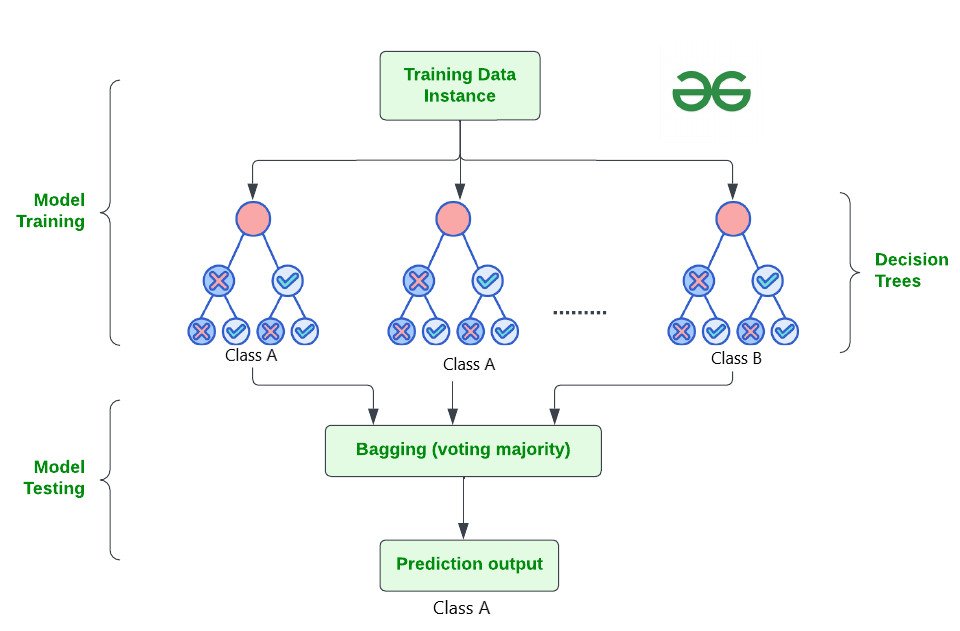


Image 2.4

* The Random Forest algorithm is a robust tree learning technique in Machine Learning. It operates by generating multiple Decision Trees during the training phase. These trees are created using the bootstrap method, where features and samples from the dataset are randomly selected with replacement to construct each individual tree.
* Within the randomly selected features, the Random Forest algorithm identifies the best splitter for classification. Similar to the decision tree algorithm, Random Forest utilizes methods like the Gini index and information gain to determine the optimal splitter. This process continues until the Random Forest builds a specified number of trees.
* Each tree in the forest predicts the target value, and the algorithm calculates the votes for each predicted target. Ultimately, the Random Forest algorithm considers the predicted target with the highest number of votes as the final prediction.

***2.2.2. How does Random Forest algorithm work?***

* Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

**Step1:** Select random K data points from the training set.

**Step2:** Build the decision trees associated with the selected data points (Subsets).

**Step3:** Choose the number N for decision trees that you want to build.

**Step4:** Repeat Step 1 & 2.

**Step5:** For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

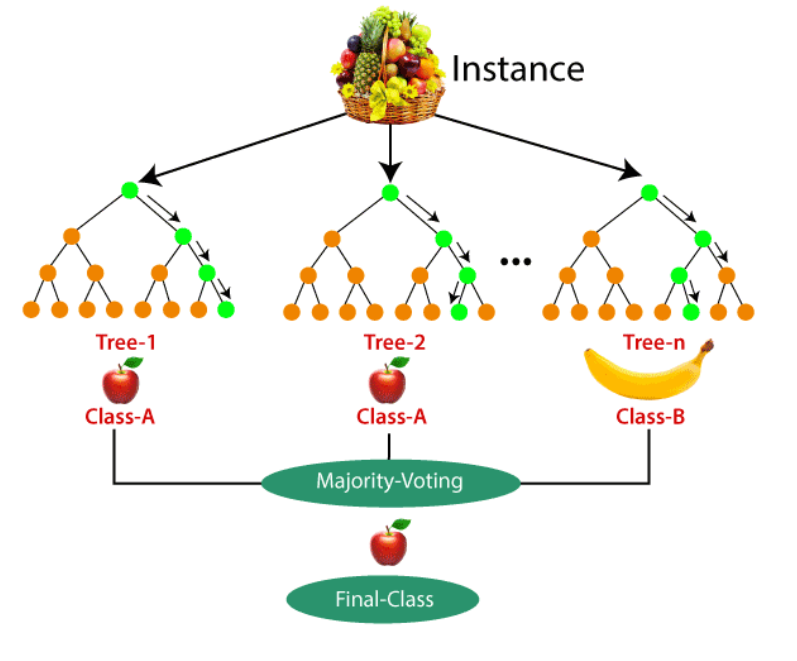


Image 2.5

* **Why use Random Forest?**

Below are some points that explain why we should use the Random Forest algorithm:

* It takes less training time as compared to other algorithms.
* It predicts output with high accuracy, even for the large dataset it runs efficiently.
* It can also maintain accuracy when a large proportion of data is missing.

## Random Forest vs. Other Machine Learning Algorithms



**2.3. SUPPORT VECTOR MACHINES (SVM)**

* SVM is a powerful supervised algorithm that works best on smaller datasets but on complex ones. Support Vector Machine, abbreviated as SVM can be used for both regression and classification tasks, but generally, they work best in classification problems. They were very famous around the time they were created, during the 1990s, and keep on being the go-to method for a high-performing algorithm with a little tuning.
* By now, I hope you’ve now mastered [Decision Trees](https://www.analyticsvidhya.com/blog/2021/08/decision-tree-algorithm/), Random Forest, [Naïve Bayes](https://analyticsvidhya.com/blog/2017/09/naive-bayes-explained/), K-nearest neighbor, and [Ensemble Modelling techniques](https://www.analyticsvidhya.com/blog/2021/09/gradient-boosting-algorithm-a-complete-guide-for-beginners/). If not, I would suggest you take out a few minutes and read about them as well.
* In this article, I will explain to you What is SVM, how SVM Algorithm works, and the math intuition behind this crucial ML algorithm.

## *2.3.1. What is a Support Vector Machine(SVM)?*

* SVM is a supervised learning algorithm used for classification and regression.
* The basic idea of SVM is to find a hyperplane in a high-dimensional space that best separates two classes of data (e.g., two classes "true" and "false").
* This hyperplane is chosen so that the distance from the closest data points to the hyperplane is maximized. These data points are called "support vectors."
* SVM is a linear algorithm in a high-dimensional space, but it can create more complex boundaries by using kernel functions (e.g., Gaussian kernel) to map the data into a higher-dimensional space.
* SVM has the ability to generalize well and is less prone to overfitting.

2.3.2. Convex Optimization and Objective Function:

* SVM finds the best hyperplane by solving a convex optimization problem.
* This convex optimization problem involves minimizing a quadratic function (typically the loss function) under linear constraints.
* Linear constraints ensure that the data points lie on the correct side of the hyperplane.
* The result of the optimization problem is the best hyperplane and the support vectors.

2.3.3. Applications of SVM:

* + SVM is widely used in text classification, image recognition, signal detection, and various other fields.
  + Particularly, SVM performs well when dealing with high-dimensional data and when the data is not linearly separable.

## *2.3.4. Types of Support Vector Machine (SVM) Algorithms*

* Linear SVM is suitable for datasets where the data points can be perfectly separated into two classes using a single straight line (in 2D). This condition of perfect linear separability means that the classes are distinctly divided without any overlap.
* Non-Linear SVM is employed when the data is not linearly separable, indicating that a single straight line (in 2D) cannot effectively separate the classes. In such cases, advanced techniques like kernel tricks are utilized to classify the data. In real-world scenarios, linearly separable datasets are rare, making non-linear SVM with kernel tricks a more common approach to handle classification tasks.

***2.3.5. Important terms***

Now, let's establish the definitions of two key terms that will be frequently referenced throughout this article:

* Support Vectors: These refer to the data points that are nearest to the hyperplane. The arrangement of these points aids in defining a separating line.
* Margin: This represents the gap between the hyperplane and the nearest observations (support vectors). In SVM, a larger margin is deemed favorable. There are two categories of margins: hard margin and soft margin, which will be elaborated upon in subsequent sections.

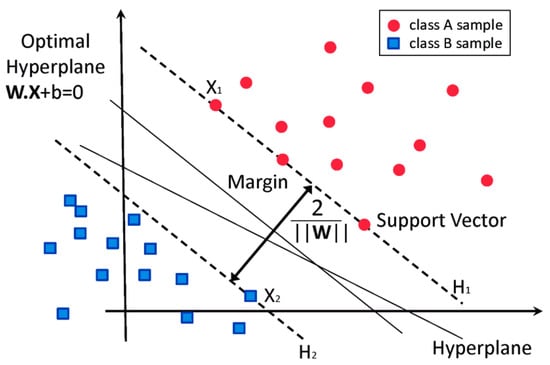


Image 2.6

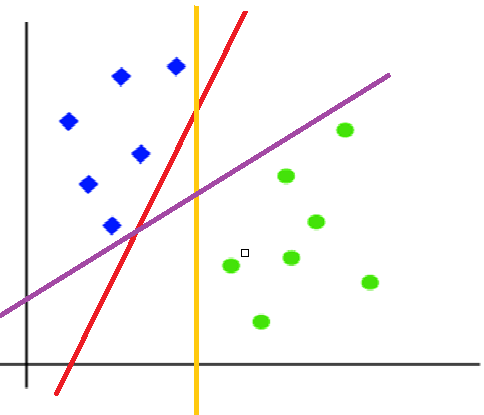
## *2.5.6 How Does Support Vector Machine Work?*

Support Vector Machine (SVM) operates by focusing solely on the support vectors, disregarding other observations since the margin is established using the points nearest to the hyperplane. This differs from logistic regression, where the classifier is defined over all points, leading to inherent efficiency advantages for SVM.

To illustrate how SVM works, let's consider an example dataset comprising two classes: green and blue. Our objective is to classify new data points as either blue or green.

  
Image 2.7

To classify these points, there can be numerous decision boundaries, but the challenge lies in determining the optimal one. It's important to note that while we depict the data points on a 2-dimensional graph, the decision boundary is referred to as a straight line. However, in higher dimensions, this boundary is termed a "hyperplane."

  
Image 2.7

The primary objective of SVM is to identify the optimal hyperplane, which is the plane with the maximum distance from both classes. This entails discovering various hyperplanes that effectively classify the labels, ultimately selecting the one with the greatest margin or distance from the data points.

  
Image 2.8

***Support Vector Machines (SVM) play a crucial role in separating phishing and legitimate websites in high-dimensional feature space by finding the optimal hyperplane that maximally separates the two classes. Here's how SVM achieves this:***

1. Hyperplane:

- SVM aims to find the hyperplane that best separates the data points of different classes in feature space.

- In a binary classification scenario like phishing detection, the hyperplane is a decision boundary that separates phishing websites from legitimate ones.

2. Maximizing Margin:

- SVM not only finds a separating hyperplane but also maximizes the margin, which is the distance between the hyperplane and the nearest data points of each class (called support vectors).

- By maximizing the margin, SVM ensures robustness and generalization ability, as it aims to find the decision boundary with the maximum separation between classes.

3. Kernel Trick:

- In cases where the data is not linearly separable in the original feature space, SVM employs the kernel trick to map the data into a higher-dimensional space where linear separation is possible.

- Common kernel functions used include linear, polynomial, radial basis function (RBF), and sigmoid kernels.

- By transforming the data into a higher-dimensional space, SVM can find a hyperplane that effectively separates the classes, even when they are not linearly separable in the original feature space.

4. Soft Margin Classification:

- In real-world scenarios, data may not be perfectly separable due to noise or overlapping classes.

- SVM accommodates such scenarios by allowing for soft margin classification, where a small number of misclassifications (errors) are tolerated to achieve a better overall separation.

- The regularization parameter (C) controls the trade-off between maximizing the margin and minimizing the classification errors.

5. Decision Function:

- Once the optimal hyperplane is determined, SVM uses a decision function to classify new data points.

- The decision function assigns a data point to one of the two classes based on which side of the hyperplane it falls.

SVM separates phishing and legitimate websites by finding the optimal hyperplane that maximally separates the classes in a high-dimensional feature space. By maximizing the margin and employing the kernel trick, SVM achieves robust classification even in complex scenarios where data may not be linearly separable.

6. Applications of SVM in Detecting Fraudulent Websites:

6.1. Binary Classification:

* + SVM is a binary classification algorithm that separates data into two different classes (e.g., fraudulent and non-fraudulent).
  + In the case of detecting fraudulent websites, SVM can be used to determine whether a website is likely fraudulent based on its features.

6.2. Learning from Training Data:

* + SVM learns from labeled training data, where websites are already labeled as fraudulent or non-fraudulent.
  + It finds the best hyperplane to separate the two classes of data.

6.3. Website Features:

* + SVM utilizes features of websites (e.g., URL, content, notifications) to determine whether a website is likely fraudulent.

6.4. Real-World Applications:

* + SVM has been widely applied in detecting fraudulent websites, email spam, text classification, and various other fields.

7. Steps for Classifying Websites as Fraudulent or Non-Fraudulent Based on Their Features:

7.1. Data Collection (Training Data):

* + Gather a dataset containing websites labeled as fraudulent or non-fraudulent.
  + Each website is represented by features such as URL length, number of links, capitalization ratio, etc.

7.2. Data Preprocessing:

* + Normalize and extract features from website URLs.

7.3. Build an SVM Model:

* + Use SVM to create a classification model.
  + SVM finds the best hyperplane to separate the data classes (fraudulent and non-fraudulent).

7.4. Model Training:

* + Train the SVM model on the labeled training dataset.

7.5. Prediction and Evaluation:

* + Use the trained model to predict whether a new website is likely fraudulent.
  + Evaluate the model’s performance using metrics such as accuracy, sensitivity, specificity, etc.

# REFERENCES

1. Shouq Alnemari, & Majid Alshammari. (2023). Detecting Phishing Domains Using Machine Learning. *Applied Sciences*, *13*(8), 4649–4649. <https://doi.org/10.3390/app13084649>
2. Dutta, A. K. (2021b). Detecting phishing websites using machine learning technique. *PLOS ONE*, *16*(10), e0258361. https://doi.org/10.1371/journal.pone.0258361
3. *Detecting Phishing Websites using Machine Learning - Javatpoint*. (n.d.). Www.javatpoint.com. <https://www.javatpoint.com/detecting-phishing-websites-using-machine-learning>
4. Alswailem, A., Alabdullah, B., Alrumayh, N., & Alsedrani, A. (2019, May 1). *Detecting Phishing Websites Using Machine Learning*. IEEE Xplore. https://doi.org/10.1109/CAIS.2019.8769571
5. Nagesh Singh Chauhan. (2020, February 9). *Decision Tree Algorithm, Explained - KDnuggets*. KDnuggets. https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html
6. *Random Forest Algorithm in Machine Learning*. (2024, February 22). GeeksforGeeks. <https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/>
7. *Random Forest algorithm — Machine Learning cho dữ liệu dạng bảng*. (n.d.). Machinelearningcoban.com.
8. https://machinelearningcoban.com/tabml\_book/ch\_model/random\_forest.html
9. Maheshwari, V. (2019, January 5). *Random Forest Algorithm*. Medium.
10. https://medium.datadriveninvestor.com/random-forest-algorithm-777e6597bfcc
11. Saini, A. (2021, October 12). *Guide on Support Vector Machine (SVM) Algorithm*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/10/support-vector-machinessvm-a-complete-guide-for-beginners/?fbclid=IwAR0bgDQXO5sm-WFKhaqVnFtZcbGAhYxrCoOo4LfgQQJfkk0wQ1evPGHc7j4