Malware detection and analysis using ML

Contents

[Abstract: 1](#_Toc121177025)

[List of Abbreviations 1](#_Toc121177026)

[Chapter 1: Introduction 2](#_Toc121177027)

[1.1 Motivations: 7](#_Toc121177028)

[1.2 Aims 8](#_Toc121177029)

[1.3 Objectives 8](#_Toc121177030)

[1.4 Artefact Description 9](#_Toc121177031)

[1.5 Research Questions 9](#_Toc121177032)

[1.6 Thesis Outline 10](#_Toc121177033)

[Chapter 2 Literature Reviews 11](#_Toc121177034)

[Static malware Analysis 11](#_Toc121177035)

[Dynamic Malware analysis using ML: 13](#_Toc121177036)

[Comparison among existing systems 15](#_Toc121177037)

[A review of artificial intelligence based malware detection using deep learning 16](#_Toc121177038)

[Gap Analysis: 17](#_Toc121177039)

[Chapter 3 Methodology 18](#_Toc121177040)

[3.1 Research Methods (Qualitative/Quantitative) Used 18](#_Toc121177041)

[3.2 Proposed Architecture 18](#_Toc121177042)

[3.3 Machine learning 19](#_Toc121177043)

[3.3 Data set: 20](#_Toc121177044)

[Various Open Source Datasets 21](#_Toc121177045)

[Machine learning proposed models 21](#_Toc121177046)

[1. EDA: 21](#_Toc121177047)

[2. Data preprocessing 22](#_Toc121177048)

[3. Data visualization 22](#_Toc121177049)

[4. Feature engineering: 22](#_Toc121177050)

[5. ML models 22](#_Toc121177051)

[References: 25](#_Toc121177052)

[Bibliography 25](#_Toc121177053)

# Abstract:

Malware detection is a crucial aspect of the information security system. Unfortunately, existing signature-based ML approaches cannot detect zero-day attacks & malware. That is why ML-based identification is required. In this research we contrast our approach to prior models for dealing with static malware analysis. We present a model that simulates the practical implementation of such a model for further investigation. We give entire sources for replicating the suggested model and its outcomes. This research includes suggested approaches for machine learning-based malware categorization & detection and implementation guidance. Furthermore, the research can be a foundation for future work in malware analysis using ML approaches.

# List of Abbreviations

|  |  |
| --- | --- |
| words | Abbreviations |
| kNN | k-Nearest Neighbor |
| ML | Machine learning |
| DT | Decision trees |
| SVM | Support Vector Machine |
| JIT | Just-In-Time |
| TN | True negative |
| AI | Artificial Intelligence |
| ROCC | receiver operating characteristic curve |
| TP | True positive |
| Malware | Malicious Software |
| OS | Operating system |
| API | Application Programming Interface |

# Chapter 1: Introduction

The amount of documented security lapses caused by viruses, trojans, ransomware, and other malware has significantly increased in recent years, with news of malware infestations making the news today more than ever. Nearly every week, one security flaw is disclosed, which might be construed as a threat (Liu, 2022).

The security community needs to manage and detect harmful content. Given a large number of cyberattacks and the popularisation and good outcome of ML strategies in classification in various domains, it is only natural to see these strategies used to supplement traditional methodologies for suspicious detection, particularly supervised training methodologies. Commonly used obfuscation strategies, for example, includes data processing such as XOR or simple character substitution, which easily hides content from novice eyes. Furthermore, some malware authors use real-time movers to conceal the dangerous software until it is downloaded into storage (Monnappa, 2015).

Viruses, malware, infections, Worms, & chatbots are all examples of malware or harmful software. Malware is designed to make money unlawfully by accessing private information, acquiring credentials stored on the system, and damaging data on the victim's computer. They are prepared to target numerous platforms such as PCs, smartphones, routers, smart televisions, etc. Malware of many types is employed in cyber espionage, computer hacking, and cyber warfare. Malware attacks can be carried out in various methods, including zip files from the fraudulent web, email attachments containing benign-looking files, and inserting malware in Word Documents or PDF documents. When a system is hacked, it is linked to other compromised computers. The botnet is an infected connection of systems. Malware typically consists of viruses engineered to move autonomously to the victim's computer by being connected to information retrieved from the internet or peer-to-peer file transfers. It may reside on a system but will not begin its attack procedure until a user clicks on an infected computer.Worms are another type of malware that spreads themselves autonomously. Its activity varies from that of a virus. Rather than attaching itself to various objects here on the system, it seeks another framework for the connection to infect after execution. Many worms really don't require social interaction to propagate (Yoshioka, 2017).

The number of malware infections globally exceeded 2.8 billion in the first half of 2022. In 2021, 5.4B malware assaults were identified. In earlier periods, the much more malware attacks were discovered in 2018, as 10.5B such attempts were registered worldwide. Figure 1 depicts the percentage of malware attacks between 2015-22 (statistic, 2022).

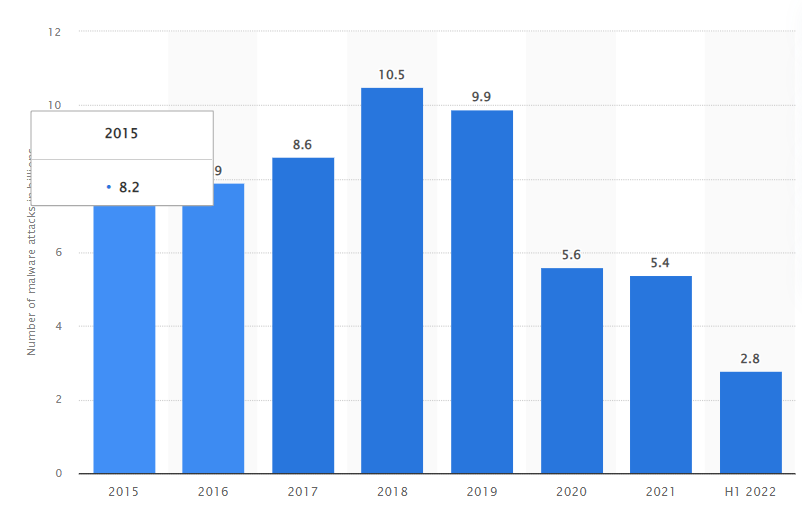


Figure percentage of malware attacks between 2015-22 (statistic, 2022).

From the Fall of 2020 until Oct 2021, the expert sector was the most targeted worldwide industry sector by malware assaults. There were 1,236 malware incidences in the industry during the monitored period. With 776 such cases, the information industry came in second. Following that, the industrial industry was the subject of 622 malware assaults.

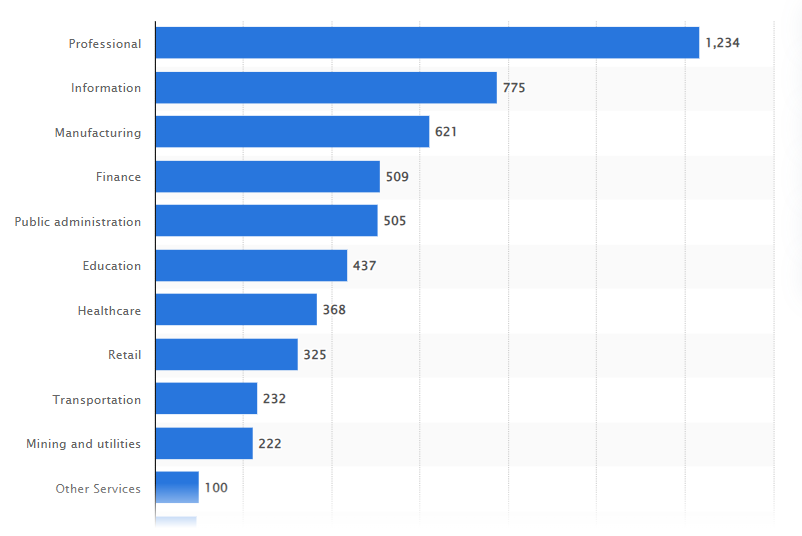


Figure malware in most targeted industry from 2020-2021

As of the beginning quarter of 2020, the malware business mainly targeted Windows computers. According to AV-Test, domestically made malware programs targeted the Windows operating system throughout the testing period.

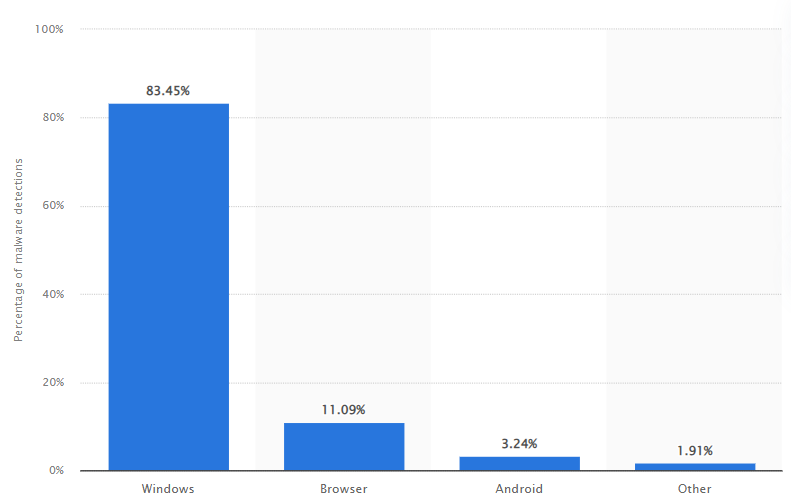


Figure Distribution of malware detections 2020 via OS

**History of Malware:**

According to several technology scientists, the first pathogen was generated around 1970. The Creeper Worm duplicated itself and copied itself throughout the ARPANET. It flashed the phrase "I'm the creeper, grab me if you can!" when enabled.

The name "virus" was not developed until 1986, when Fred Cohen, a PhD student, defined a malicious program as a programme that may infect other programmes and build an evolved edition of itself. The majority of early viruses either deleted data or infected booting sectors. Nowadays, malware is far more dangerous, meant to install malware, spy on institutions, cause a denial-of-service problem, or freeze files to extract money from users.

**Types of Malware:**

Data breaches continue to evolve in complexity as businesses and organisations deploy newer systems and interact more deeply with cloud solutions. While the intricacy of the attacks grows, they frequently employ the same underlying tactics. One of the most prevalent techniques for gaining access to sensitive information is deploying computer viruses or malicious programs to a system.

These various malware programmes then steal, erase, or take control of critical data and functions. Even though most individuals know that computers may get infected, many cannot identify at least three categories of infection or correlate the hazards associated with each type. Today, we'll look at the many sorts of malware and how to avoid ransomware attack programes.

The following are some of the most popular malware kinds and evidence:

**1. Worms:**

Worms infiltrate a computer's memory and multiply to control an entire network. When the software is installed, it will run, allowing it to change and remove data, inject other forms of malware into the system, bottleneck infrastructure elements, and extract personal documents.

Among the most frightening elements of a crawler is that it is a sort of software that multiplies without the involvement of humans(worm, n.d.).

**2. Trojan Horses:**

Trojan Horses, as the name indicates, are malicious schedules that pose as harmless files while concealing their actual nature beneath. These virus varieties use their deception to trick users into downloading what looks to be a legal file. Once inside, hackers can use them to transmit other types of viruses or to construct backdoors into the system.

**3. Spyware:**

Spyware is a sort of malware that typically disguises itself as adware. These malicious apps are used to secretly acquire user data and passwords. Spyware-infected computers may unwittingly convey crucial data to hackers, including credit card info. They also act as a gateway for possible ransomware attacks(pyware, n.d.).

Because of its covert nature, malware may be extremely damaging. As a result, many businesses must learn how to evaluate ransomware security (Mohanta, 2020).

**4. Virus:**

Ransomware is a malware attack that encrypts data on the cloud and prevents users from accessing such files or systems. Then, utilising encryption, all files, or maybe even entire equipment, are kept hostage again until the victim expends a ransom in payment for a decryption key. The key allows the client to access the network's encrypted data or systems**.** (Anon., n.d.).

**5. Ransomware**

Ransomware tries to take over a computer but locks the user out by requesting a ransom (ransomware, n.d.). Ransomware, like a worm, may swiftly reproduce and contaminate multiple terminals over a connection and, if successful, can encode all files on a device, rendering it inoperable.

Ransomware frequently seeks money in the form of cryptocurrencies in return for decrypting files on the system, although it does not always do so after the payment.

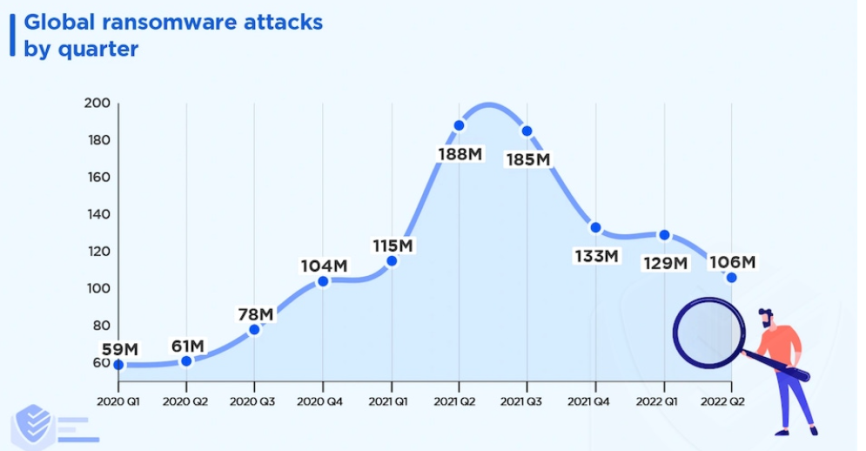


Figure Global ransom ware attacks statistic

**6. Adware**

Measures to expose the user to potentially damaging and unwanted advertising. Even though adware is harmless, the advertising that appears and is accessed by the user may contain malware. Adware's damage is often easier to remediate than other malware varieties (adware, n.d.).

# 1.1 Motivations:

Malware, often known as malicious software or code, is characterised as "any code introduced, updated, or removed from a software system to purposefully inflict harm or disrupt the system's intended function". Malware could be a software component connected to a legal application, a separate programme, or a mix of the two.

It is generated for various reasons; some are planned to show a vulnerability or idea that does not cause physical harm to networks, while other reasons are used to extract identities or render systems worthless. The sheer variety and number of today's systems and the amount of classified info they contain present various chances to benefit unlawfully through compromising authorised networks.

Since the 2000s, the number of malware has grown exponentially, from 1 mm in 2007 to over 500 million by the end of 2019. According to a McAfee1 analysis from 2009, around 4% of powerful search company query results link to potentially harmful websites. According to a more recent study by Symantec in 2022, at least 10% of the latest trending search queries generated damaging effects.

In 2019 alone, 1.1 billion identities (namely user personally identifiable information) were disclosed due to encounters with malicious programs.

Anti-virus technologies rely on two major ways to identify malicious software: signature-based, which utilises a database of existing malware to detect malware, and heuristic-based, which employs harmful tendencies and a set of guidelines to detect either known or novel malware.

When dealing with the problem of identifying both new and old malware, the application of ML techniques has shown promising results, but not without its drawbacks.

The first stumbling block is the lack of an exhaustive review of malware; not only does the lack of an accepted definition result in different anti-virus resellers classifying the same programme differently, but some programmes fall within a grey area for which no clear characterization can be judged correct. Adware, or advertising-supported software, is one kind that, although not explicitly malevolent, performs presumably unrequested acts. The lack of precise measurements and attributes that distinguish ransomware from good necessitates extra work in developing datasets for malware detector assessment.

Another issue while using ML for malware detection is determining how to analyze the learning approach. Whereas many classification problems are time-independent, malware detection may not be. As a result, using conventional assessment procedures that neglect any temporal consistency affects model performance.

# 1.2 Aims

This research uses machine learning to investigate the public dataset to learn more about the malware's operations, behaviours, and activities on the victims' systems.

# 1.3 Objectives

This research's main objectives are as follows:

* It offers a novel ML-based malware detection technique .
* It offers the testing and training of the given model using multiple ML techniques.
* It also emphasises the effectiveness of the suggested technique in the light of faked malicious attack

# 1.4 Artefact Description

• A method for using machine learning to identify and categorize targeted malware based on behavioral and memory data.

• Describe the analysis process, procedural phases, characteristics that were deleted, and ML algorithms.

• Provide refreshed memory components resulting from memory analysis and new properties. All memory parts are comprehensive and thoroughly discussed.

• Thorough study of the given technology, proving that a careful malware detection strategy yields excellent accuracy and a low false-positive rate.

# 1.5 Research Questions

To answer some of the research questions, this research will look to existing literature. Some of the RQ associated with this research that will be answered by the literature review are:

RQ1: Why perform Malware analysis ?

RQ2: How does malware Spread?

RQ3: How do we evaluate the Prediction Performance of malware Detection solutions?

RQ4: Which machine learning algorithms should we use?

RQ5: How do we explore the Practical Potential of malware Detection?

# 1.6 Thesis Outline

Chapter 2 Literature Review. We present the concept of ML and its applications, observed by related work that specifically uses ML to detect malware.

Chapter 3 Methodology: This describes our initial process & procedure to help solve the malware detection problem using ML. We detail the data-gathering process and how it was analyzed, followed by the labelling process and dataset creation.

Chapters 4 and 5 Implementation and results take the created datasets to build a malware detection model based on features and how these perform under different validation methodologies.

Chapter 6 Conclusion and future work

# Chapter 2 Literature Reviews

This chapter examines preliminary research and exposes problems in their techniques, and highlight gaps.

Malware analysis is an integral part of the cyber security process. Machine learning techniques for static or dynamic analysis have been widely analysed in research in recent years. Since malware makers' goals and objectives have shifted, malware is evolving in its shape and infection tactics, not simply for celebrity, political espionage, or financial gain. Targeted malware is a relatively new type of malware that has received little attention.

## 2.1 Static malware Analysis

We present a synopsis of current work exploiting machine learning infosec modelsusing static ML approach.

In this (Balram, 2019), The authors discuss the creation and deployment of 6 machine learning models, as well as two unique kinds of intrinsically derived features from executable files: strings and Portable Compiled code suggests that demand in this work. To function for each feature category individually, each of the six classifications received a total of twelve malware detectors. The scikit-learn ml toolkit was used to create these classifications & methods for feature extraction in Python. The detection efficiency & batch processing requirements of the twelve malware analyzers were evaluated & studied.

In this (Nazareno, 2021), Every network traffic collected by each malicious sampling is used to evaluate the behaviour of nine different malware variants dynamically. This conduct was then contrasted to those seen in physical surroundings. The findings show that viral activity differs considerably across these two environments.

In this (ASLAN, 2017), The authors used multiple static malware analysis tools and antivirus scanning programmes studied in this work. In a test example, 200 malicious or benign files were gathered from various sources & examined on different Windows versions. Once opposed to static analysis methods, test findings reveal that antivirus software detects existing malware faster and more efficiently. Furthermore, for unknown malware, static analysis technologies outperformed virus protection.

(Anderson, 2017), Authors summarise the numerous attacks suggested models of machine learning in cybersecurity, each requiring the adversary to know something about the algorithm under attack. Notably, even when used to target a machine learning malware classification dependent on static attributes for Desktop portable executable files, earlier attack approaches may damage the malware's format or operation. They examine a more generic reinforcement learning-based architecture for assaulting static anti-malware engines, miming more realistic attacker situations and hence giving considerably lower detection rates.

A reinforcement learning agent has a set of functionality-preserving procedures that it may use on the file. It determines which sequence of actions will lead to evasion for a specific malware piece by playing a series of games against its anti-malware algorithm. Given the overall architecture, it is unsurprising that evasion rates are low. Furthermore, the resultant RL agent may quickly characterise the pro model's blind spots. Furthermore, the agent's deceptive variations may be leveraged to improve ml anti-malware engines through transfer learning. As illustrated in below figure, authors use deeper Q-learning in an RL framework to evade malware.

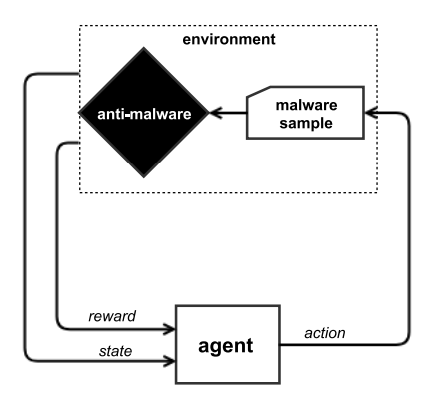


Figure reinforcement learning problem

## 2.2 Dynamic Malware analysis using ML:

Dynamic analysis is the technique of studying a binary file by running it in a contained way & observing its behaviour to determine whether or not it is dangerous.

(Jamalpur, et al., 2018), The authors showed the use of sandboxing to recognize authorized malware coding fragments in 2 methods and then assess their behaviour by evaluating the virus's activities in the cuckoo environment. Cuckoo is a malicious code investigation group that does in-depth malware analysis and provides comprehensive results that have been tested.

(Han, 2019) , findings of the author's experiments demonstrate a fundamental link between malware's dynamic & static API call patterns. "The syntax differs, however the ideas are comparable," summarises the connection. This study first explores the differentiation and link between dangerous application static and dynamic API sequences based on this discovery. Before generating the hybrid feature vector space, they link and fuse their stochastic and dynamic API events into a single hybrid sequence based on the semantic mapping. Furthermore, we mine and classify the many types of malicious behaviour of the programmes and provide comprehensible data for malware detection. This study analyzes an issue with previous strategies in that they frequently focused on identification but ignored explanation. They develop MalDAE, a responsive malware detection method, by correlating and merging static & dynamic API events. Based on the assessment results, the detection and accuracy rate may reach 97 to 94%, respectfully outperforming prior similar studies in a thorough comparison. Additionally, the authors outline common types of malware and proactively assist in identifying and combating malware.

The authors this (Zhou, 2020) describe a method for categorising populations that employ the regression equation of API groups as the SVM feature values. The correlation coefficient between API groups was chosen as the feature quantity because various ransomware families display diverse behaviour patterns, which the association between API groups may suggest. According to the results of an evaluation experiment, the proposed approach achieved a 99.9 per cent accuracy, suggesting that the subspecies were correctly classified. Otherwise, the analysis of API contributions demonstrates that each API's contribution to classifying ransomware groups varies.

Author (Taheri, 2019) used the second part of the publicly available example, which includes permissions & objectives as static features and API calls as dynamic features. They also utilise our two-layer Android malware scanner to go through these features. According to their conclusions, they obtained a 96 recognition rate in Static-Based Malware Binary Identification, 83 per cent reliability in Dynamic-Based Malware Category Classification, and 59.7 per cent accuracy in Dynamic-Based Malware Family Classifier in the first layer.

(Saad, 2019) applied ML approaches and have yet to be ready to identify malware in the environment. They feel dynamic detection methods are the way to go in today's world for malware detection implementation of security systems due to the current trend in malware generation and the growth in atypical malware attacks. A comprehensive evaluation of deep learning for attack detection is offered. Then they discuss how malware detection poses new challenges for history's cutting-edge ML algorithms. They discovered three significant difficulties that limit the effectiveness of ML-powered detecting attacks in the environment. Following that, we will discuss potential solutions to these issues and the requirements for the subsequent detection of assault. Finally, they discuss potential machine learning future research for identifying assaults.

## 2.3 Comparison among existing systems

To finalize the literature survey, multiple studies were chosen from the total number of publications that studied performance appraisals and confirmability. Table 2.1 shows a few intriguing strategies with good performance discovered from these studies. Because static analysis is the approach on which this research is based, the content is exclusively static analysis.

Table 2.1 Table of comparison of Previous research

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Authors** | **Method** | **Limitations** |
| 1 | (Aggarwal, 2022) | implementing several categorization models such as Logistic Regression, Neural Network, and SVM | limited predictions when train on low dataset. |
| 2 | (Ravi, 2022) | J48, JRip, PART, Random ForestRF , Naive BayesNB, Logistic, & RIDOR | Based on the representation methods utilised by those models to understand the properties of both classes, the suggested models have become capable of differentiating between malware and benign files. |
| 3 | (Salah, 2022) | use 15 classes in the data, therefore we used tree-based machine learning methods. Random-Forest RF  and Xgboost XGB models were proposed. | it takes more time to analysis |
| 4 | (Gibert Llauradó, 2022) | deep learning to extract N-gram like features from the assembly language instructions | forces certain text classification algorithms to have a poor detection rate |
| 5 | (Serpanos, 2021) | SVM, RF, Decision tree | High dimensionality space |
| 6 | (Chaganti, 2021) | Convolutional Neural Network (CNN) Image based model | describe the features makes the features subject to obfuscation methods that generate outliers such like unnecessary API calls or opcodes without compromising proper operation. |
| 7 | (Demirezen, 2021) | multimodal CNN-based deep learning structure and picture feature extraction approach based on singular value decomposition are offered. | reduce the amount of extracted characteristics by retaining just the most frequently recurring ones |
| 8 | (Majid, 2021) | CNN, RNN, LSTM and auto encoders. LSTM | Graph-based models to have a significant level of complexity over time. |
| 9 | (Xiao, 2021) | CoLab, VGG16, and SVM is proposed. | time matching has been examined |
| 10 | (Mijwil, 2020) | Random Forest (RF), support vector machine (SVM), *NaïveBayes (NB)* and decision tree (ID3) are applied | Take lot of time in training |
| 11 | (Ahmed Amer, 2019) | Gradient Boosted Decision Tree model | The model only looks at questionable items that are not on either the recognized whitelist or the banned blacklist. |
| 12 | (Rathore, 2018) | used opcode frequency as a feature vector and unsupervised learning in addition to supervised learning. | visualization  that the extracted characteristics are saved as photos, which need additional space to store. |

## 2. 4 Gap Analysis:

Throughout the literature study, it was clear that a significant amount of research had been conducted on this topic. Given the various approaches presented, it's difficult to tell which is the most effective. Even when researchers refer to comparable works and compare the results, it is hard to determine whether there is a distinction in results owing to various uncontrollable circumstances. The dataset must be the most crucial component among these changes since it can substantially alter the outcomes of a technique.

Taking a step back and investigating state-of-the-art approaches, this thesis can assist in determining the distinctions between the proposed methods and assessing whether there is a significant difference between them.

To remove a research gap, this thesis first compares and contrasts standard ML architectures for malware detection, classification, and categorization uses open-source public and private datasets. Second, we eliminate all dataset bias removed in the experimental analysis by employing disjoint divisions of the public and private datasets to train and test the model on various timelines.

This chapter gave context and explored various connected concepts. It began by outlining the various forms of malware analysis (i.e. static & dynamic analysis). It also examined the primary malware detection approaches (signature-based and feature-based) and the many strategies used by these detection methods. The chapter also examined various backgrounds and terminology required to follow the work or suggestions, such as ML technology.

# Chapter 3 Methodology

This chapter describes the current methodology and the modifications required for this investigation. The analytic system and the Machine Learning methods utilized in this research are then introduced. The following Section will present a state-of-the-art study on malware detection and categorization.

## 3.1 Research Methods (Qualitative/Quantitative) Used

A quantitative research technique is applied in this study. The primary goal is to provide users with a secure real-time facility for identifying Phishing URLs. Furthermore, an experimental inquiry is conducted using a Malware detection system. Furthermore, a literature-based study is performed to gain essential knowledge about the topic, collect background facts, & analyze gaps.

## 3.2 Proposed Architecture

Machine learning employs a variety of methodologies. Unsupervised and supervised machine learning is two separate forms of machine learning. Supervised learning comprises a training step in which a model is fitted to training data and then used to forecast. As a machine learning approach, deep learning employs the notion of supervised learning to learn and predict outcomes using layers of artificial neural networks. It has been demonstrated to be beneficial for object identification and picture categorization. Linear regression, Random forest, Decision tree and KNN are machine learning algorithms used to aid in malware detection and analysis.

  The methodology will be heavily influenced by the answers to some of the research questions in chapter 1. Among these are the machine learning algorithms used in development.

The flowchart below depicts the suggested actions that will be implemented throughout the development in this research.

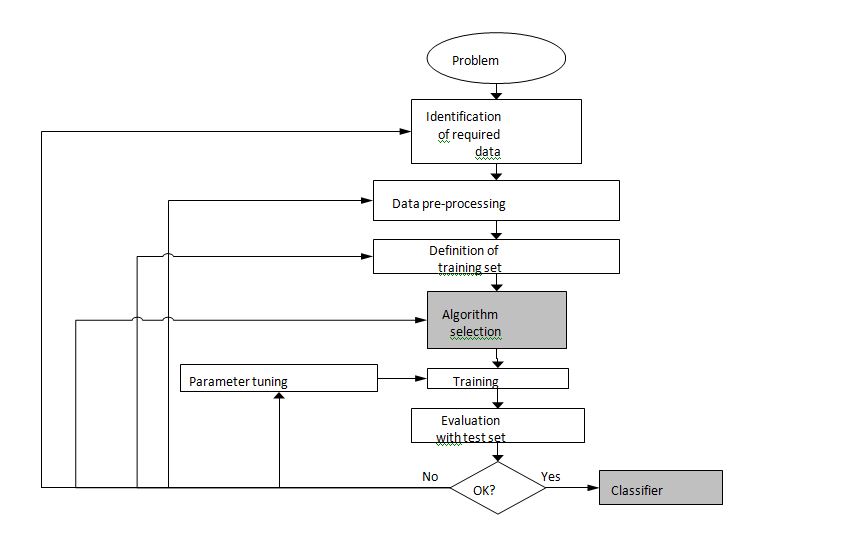


Figure Flow Chart for development

## 3.3 Machine learning

Before Alan Turing created the current notion of computers, humanity had constantly sought ways to simplify or assist more complicated processes. Computer science topics such as AI have considerably advanced in recent years due to technological developments that enable greater processing capacity.

ML is one of the AI domains that has benefited from technological advancements. The next part will offer a more extensive discussion of ML or its application to the current task.

A critical distinction between the current definition and the ML field is that those algorithms are not meant to solve specific issues. If a learning problem is clearly described by T, P, & E, an approach utilizing these three characteristics can learn but only sometimes solve any problem. This contrasts with AI areas, where algorithms are developed for specific purposes.

After specifying a learning issue, one or more learning algorithms are used to provide a meaningful conclusion (i.e., model). Learning algorithms are classified into four types: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.

* **Supervised Learning SL:**

SL algorithm is given a set number of instances (inputs) to study from, and these examples are labeled; they are associated with the desired outcome (outputs). The algorithm is then tasked with determining the law that allows it to determine the output based on the inputs. The goal is to find the optimum function f(x) capable of connecting the intake (x) toward the output (y).

Two primary categories of issues may be handled using supervised learning: classification difficulties and linear discriminant analysis.

* **Unsupervised learning USL:**

Neither label is supplied to the algorithm, which finds the input's distinctive structure without humans' intervention. The machine will create its representation, which a human may find challenging to comprehend. To construct homogenous groupings from the observations, common patterns are found. Unsupervised learning is further subdivided into clustering and relationships. The aim behind clusters is to look for commonalities in data to build groups. Association algorithms, as opposed to clustering, are responsible for detecting rules in data. These rules can be expressed as "If circumstances X and Y are satisfied, occurrence Z may occur."

* **Semi-supervised SSL:**

This group includes the preceding ones and arises because the learning challenge is occasionally based on a sample with both unlabelled entries or a collection with complex data. In such cases, combining supervised or unsupervised learning yields a superior overall outcome.

* **Reinforcement Learning RL:**

In this class, the program is given a task to solve using a set of conditions and actions. At any point, the software does an action that results in an incentive or penalty. Each state's progressive reinforcement trains the program for the best and worst behaviors for that state.

## 3.3 Data set:

A dataset is a massive set of information gathered to train or forecast a classification. Datasets for malware identification are frequently gathered from malware archives such as VirusShare, VX Heaven, & Mal share. Unfortunately, malware obtained on these sites may be obsolete, but because malware constantly evolves, it is critical to train classifiers on fresh samples. Some other option is to use a honeypot to catch live malware. This represents a purposefully susceptible server designed to detect computer breaches or, in this example, to catch malware installed on the system. This might be a time-consuming activity that demands careful updating.

### Various Open Source Datasets

Datasets for ML can be generated privately or made publicly available to anyone or anything. This section will explain what prominent datasets are available and how they are often utilized in the study.

* In 2015, Microsoft launched "The Microsoft Classification Competition", a malware classification challenge with a dataset of 0.5 terabytes. This dataset has grown in popularity among academics and is frequently used to compare malware classification algorithms.
* Ember (embre, 2020) is a substantial public dataset including static characteristics collected from PE files. The dataset contains 1.1 million binary file characteristics and may be utilized for testing and evaluating different virus detection strategies.
* Malimg is an image dataset created by converting malicious binary files. This greyscale picture collection contains 9,458 samples from 25 distinct malware families.

## Machine learning proposed models

The Machine Learning technique entails creating a Predictive model that can be utilized to solve a Problem Statement. The goal is to use Machine Learning to detect malware. In Malware detection using the Machine Learning process, the following stages are taken:

### 1. EDA:

In this part, we will go through many machine-learning methods that may be applied to malware detection and do an exploratory data analysis.

While examining and researching data sets, data scientists typically use exploratory data analysis (EDA), summarising the significant aspects of the data to the visualization approach. It helps the Data Scientist identify Data Patterns, spot anomalies, test hypotheses, and make assumptions.

In a word, it is a method that helps the Data Scientist choose the best ways to transform the provided data source to get the intended outcome.

Based on our analysis and observations, we make future decisions about which features to include in the dataset when training and which to exclude to improve the model's accuracy.

#### 1.1 Data description:

Malware is a well-organized, well-funded market dedicated to circumventing standard security solutions. Once a computer has been infected with Malware, thieves may use it to harm consumers and businesses in various ways.

Microsoft, which has over one billion commercial and consumer customers, takes this issue extremely seriously and is heavily involved in strengthening security.

In keeping with their larger goal, Microsoft is pushing the data science community to create algorithms to anticipate if a system will be infected with Malware shortly. Microsoft is giving Kagglers an unparalleled malware dataset, like with their last Malware (2015), to stimulate open-source advancement on experimental approaches for forecasting malware occurrences.

During the "Microsoft Malware Classification Challenge" on Kaggle in 2015, Microsoft made a large dataset available to the public. The dataset contains 20K harmful samples from nine families, which are provided in binary and assembly (.asm) format and have been disassembled using the IDA Pro found different.

### 2. Data preprocessing

Before we begin training, we must first pre-process the data. This may include EDA to determine how variables or samples interact. For example, we may examine the association between the response variable and retain only one variable out of that group. Furthermore, specific training methods may be sensitive to data scales or outliers. We should address those concerns at this stage. The data may contain missing values in some circumstances. We can either discard the samples with missing values or attempt to blame them. Numerous machine learning methods are incapable of dealing with incomplete data.

### 3. Data visualization

Data visualization is critical in the display of little and big-scale data. One of a data scientist's most important abilities is telling a captivating story while displaying data and conclusions understandably and entertainingly.

### 4. Feature engineering:

Malware dataset traits and attributes are represented as structural columns in this input. To perform successfully, all algorithms require data characteristics with specified properties. Feature engineering aims to build an input dataset that fits the requirements of machine learning models. Finally, we begin by turning all categorized features into comparable number labels.

### 5. ML models

Given the nature of the task as a classification issue, numerous models may be used to describe the goal function.

We use Supervised learning and divided the data in 80:20%, which consists of;

1. Train a model & fitting a model to available training data.
2. Applying the prepared model to unique samples & acquiring predictions

This training data is used throughout the training phase to find the optimal model that will provide the proper label Y for previously undiscovered objects given the feature set X.

In the event of malware identification, X might represent specific file content or activity characteristics, such as file statistics or a list of API calls called. Labels Y might be malware or innocuous or more specific, including a virus, Trojan-Downloader, or adware.

During the training step, we must choose a model family, such as decision trees, linear regression, random forest and Knn. Its parameters often determine each model in a family.

Training entails searching for the model from the chosen family with a particular set of parameters that provides the best correct responses for the trained model over the collection of reference objects using a specific measure. In other words, we 'learn' the best parameters for defining a valid mapping from X to Y.

We can now apply the model to new objects once we have trained it and validated its quality. The model's type and parameters do not change throughout this phase. Only the model predict accuracy.

We must employ an efficient learning approach to find a model with a high detection rate or a low false positive rate.

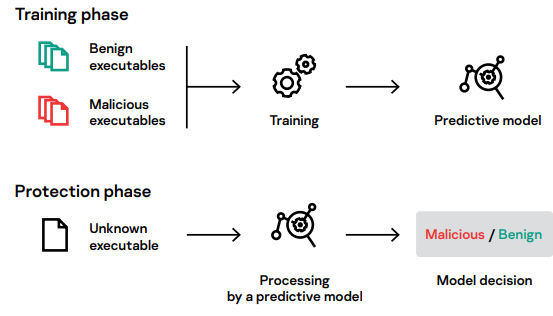


Figure ML detection life cycle

The models described below were taken into account during implementation.

#### 1. Logistic regression

Logistic regression is a statistical model used to assess if an independent variable influences a binary dependent variable. This indicates that, given input, there are only two possible results. For example, it might be used to assess whether or not an email is spam by looking at the frequency of misspelled terms, which is a frequent symptom of spam. Other logistic regression types, such as linear regression, need to establish a threshold to differentiate between binary groups (e.g., 50% misspelled = not spam, >50% misspelled = spam). Linear regression can provide a probability, but it must then be used as a logistic regression to generate a different categorization.

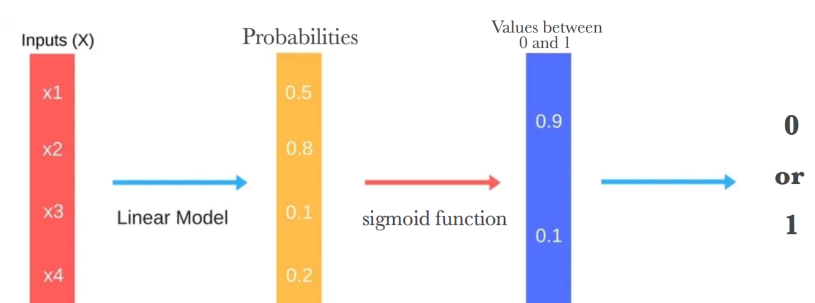


Figure linear regression

A sigmoid function is a popular model. The sigmoid function, often called a squashing function, limits outputs to values between 0 and 1. In this case, we may apply the model:

y = e^(b0 + b1\*x) / (1 + e^(b0 + b1\*x))

there's many variables b0, then b1. These are referred to as the weights or coefficient values. B0 indicates the bias, or intercept, and b1 is the coefficient. The underlying data set is used to learn and train these weights. This formula will yield a percentage or probability projected over discrete classifications. The classification model is the determined distinction between two classes. For illustrate, if a probability is more than or less than a given threshold, it falls into one of two categories.

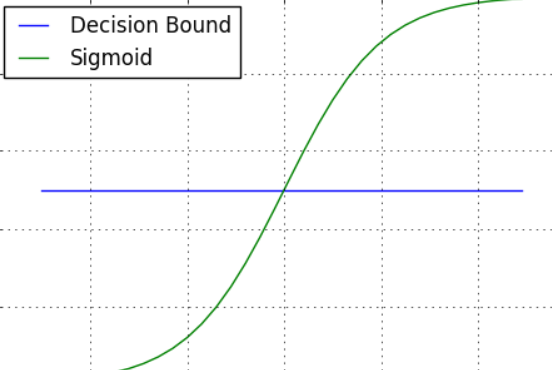


Figure linear regression sigmoid

#### 2. Random Forest

Random forest is a strategy based on decision trees used in modeling prediction and pattern analysis. It contains numerous tree structures, each reflecting a unique instance of the random forest's classification of entering data. The random forest approach examines each incident independently, selecting the forecast with the highest votes (Team, 2022).

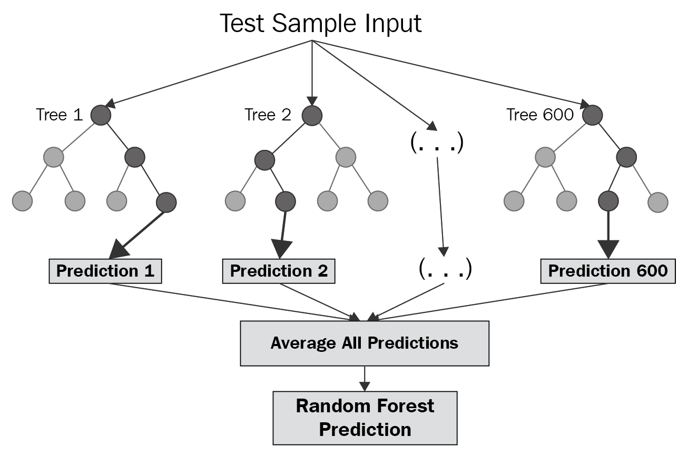


Figure Random forest

Every tree in the classifier is fed observations from the original set. The features are then chosen randomly and used to build the tree at each node. Every tree in the forest should be clipped at the end of the session when the forecast is achieved conclusively. As a result, the random forest allows any classifier with weak correlations to produce a robust classifier.

#### 3. Decision Tree

These approaches simulate discrete-valued target functions using a decision tree as the learned function. The decision tree is constructed depending on the amount of information each characteristic provides (i.e., information gain). These are best suited to issues in which examples are expressed by descriptor pairs of discrete values and problems in which data may contain mistakes. The drawbacks of decision tree approaches include how deep the decision tree should be grown, how to handle ongoing attributes, and how to manage attributes with various weights.

It may be used for classifying as well as regression. The algorithm divides the input into smaller groups and then makes predictions using these parts. A decision criterion, such as data purity or entropy, determines each split. The decision tree algorithm will continue to separate the data until it can no longer enhance the predictions. The tree is believed to have "grown" at this stage. Both category and numerical data may be used with the decision tree technique. It is a commonly used algorithm since it is simple to comprehend and apply to many problems.

#### 4. KNN

The K-Nearest Neighbor classification is the primarily supervised classifier that any data science practitioner should be familiar with. This approach was initially used for a pattern categorization problem by Fix and Hodges in 1951. The word KNN classifier was chosen to be comparable. KNN is designed to do pattern recognition jobs.

K-Nearest Neighbor, or KNN, is a supervised learning method that may be used to solve regression and classification issues. It is commonly used in machine learning for classification challenges.

# References:

# Bibliography

Abd Majid, M. a. K. A. Z. A., 2021. "Model for successful development and implementation of Cyber Security Operations Centre (SOC).. *PloS one 16, no. 11.*

ADFA, U. C. a., 2015. *The UNSW-NB15 Dataset.* [Online]   
Available at: https://research.unsw.edu.au/projects/unsw-nb15-dataset

admin, 2019. [Online]   
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

admin, 2020. [Online]   
Available at: https://metamind.readme.io/docs/use-feedback-to-improve-a-model

admin, 2020. [Online]   
Available at: https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/

admin, 2020. [Online]   
Available at: https://hbr.org/1964/07/decision-trees-for-decision-making

admin, 2021. *malware-os-distribution/.* [Online]   
Available at: https://www.statista.com/statistics/680943/malware-os-distribution/

admin, 2022. *malware-infection-weekly-industries/.* [Online]   
Available at: https://www.statista.com/statistics/223517/malware-infection-weekly-industries/

adware, n.d. [Online]   
Available at: https://www.kaspersky.com/resource-center/threats/adware

Agarwal, A. K. L. R. R. G. T. T. S. a. P. K. S., 2021. Honey Encryption: Fortification Beyond the Brute-Force Impediment. *In Advances in Mechanical Engineering, pp. 673-681. Springer, Singapore.*

Aggarwal, N. P. A. a. R. G., 2022. Static Malware Analysis using PE Header files API.. *In 2022 6th International Conference on Computing Methodologies and Communication (ICCMC), pp. 159-162. IEEE.*

Agyepong E, C. Y. R. P. B. P., 2020. Challenges and performance metrics for security operations center analysts: a systematic review.". *Journal of Cyber Security Technology 4, no. 3 (2020): 125-152..*

Agyepong, E. Y. C. P. R. a. P. B., 2020. Towards a framework for measuring the performance of a security operations center analyst.. *In 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security).*

Ahmed Amer, N. A. A., 2019. Malware Detection through Machine Learning Techniques. *International Journal of Advanced Trends in Computer Science and Engineering.*

Al-Abassi, A. H. K. A. D. a. R. M. P., 2020. An ensemble deep learning-based cyber-attack detection in industrial control system. *IEEE Access 8.*

Al-Abassi, A. H. K. A. D. a. R. M. P., 2020. An ensemble deep learning-based cyber-attack detection in industrial control system. *IEEE Access 8.*

Alahmadi, B. A. L. A. a. I. M., 2022. 99% False Positives: A Qualitative Study of SOC Analysts’ Perspectives on Security Alarms. *In Proceedings of the 31st USENIX Security Symposium (USENIX Security), Boston, MA, USA,.*

Alexandropoulos, S.-A. N. S. B. K. a. M. N. V., 2019. Data preprocessing in predictive data mining.. *The Knowledge Engineering Review 34 .*

Al-Hashedi, K. a. M., 2021. 2021. Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019.. *Computer Science Review, 40, p.100402..*

Ali, L. C. Z. Z. Z. a. Y. L., 2019. Automated detection of Parkinson’s disease based on multiple types of sustained phonations using linear discriminant analysis and genetically optimized neural network.. *IEEE Journal of Translational Engineering in Health and Medicine 7 (.*

Anderson, H. S. A. K. B. F. a. P. R., 2017. Evading Machine Learning Malware Detection.

Andrade, R. a. J. T., 2018. Enhancing intelligence SOC with big data tools.. *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE.*

Anon., n.d. [Online]   
Available at: https://en.wikipedia.org/wiki/Computer\_virus

Anon., n.d. [Online]   
Available at: https://www.kiamotorssouth.com/services-parts?gclid=CjwKCAjwqauVBhBGEiwAXOepkVYvASDtbBlJBmFGoax\_bfXl9ZaX7tZMgOyOgZ2UDcbdtt0I-KhlShoCCVcQAvD\_BwE

Arun Kumar, S. R. R. R. R. a. R. R., 2021. A survey on graphical authentication system resisting shoulder surfing attack.. *In Advances in Artificial Intelligence and Data Engineering, pp. 761-770. Springer.*

ASLAN, Ö., 2017. Performance Comparison of Static Malware Analysis Tools Versus. *International Multidisciplinary Studies Congress (IMSC), Akdeniz University, Antalya/Turkey on.*

Aung, W. P. H. H. L. a. K. K. L., 2020. Developing and analysis of cyber security models for security operation center in myanmar.. *In 2020 IEEE Conference on Computer Applications (ICCA).*

Avainash, 2018. [Online]   
Available at: https://www.datacamp.com/tutorial/k-nearest-neighbor-classification-scikit-learn

Balram, N. G. H. a. C. M., 2019. static malware analysis using machine learning algorithms on apt1 dataset with string and pe header features.. *In 2019 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 90-95. IEEE.*

Basit, A. M. Z. X. L. A. R. J. Z. J. a. K. K., 2021. A comprehensive survey of AI-enabled phishing attacks detection techniques.. *Telecommunication Systems 76,.*

Basyurt, A. S. J. F. P. K. M.-A. K. a. M. M., n.d. Help Wanted-Challenges in Data Collection, Analysis and Communication of Cyber Threats in Security Operation Centers. *2022.*

Bellovin, S. M. a. M. M., 1992. Encrypted key exchange: Password-based protocols secure against dictionary attacks..

Benedek, B. C. C. a. B. Z. N., 2022. Automobile insurance fraud detection in the age of big data–a systematic and comprehensive literature review.. *Journal of Financial Regulation and Compliance.*

Beulah M, P. M. B., 2022. Detection of DDoS Attack Using Ensemble Machine Learning Techniques.. *InSoft Computing for Security Applications .*

Bhowmik, R., 2011. Detecting auto insurance fraud by data mining techniques. *ournal of Emerging Trends in Computing and Information Sciences 2, no. 4 (2011):.*

Bi, Q. K. E. G. J. K. a. J. L., 2019. What is machine learning? A primer for the epidemiologist.. *American journal of epidemiology 188, no. 12 (2019): 2222-2239..*

Bryant, B. D. a. H. S., 2017. A novel kill-chain framework for remote security log analysis with SIEM software.. *computers & security 67 .*

Chaganti, R. V. R. a. T. D. P., 2021. Image-based malware representation approach with EfficientNet convolutional neural networks for effective malware classification.. *Journal of Information Security and Applications 69 (2022): .*

Chen, Q. S. R. I. H. H. a. R. A. B., 2019. Automated ransomware behavior analysis: Pattern extraction and early detection.. *In International Conference on Science of Cyber Security, pp. 199-214. Springer, Cham.*

Cloud-Based, M. D. A. P. A. T. i., 2019. Hou, Jiahui and Qian, Jianwei and Wang, Yu and Li, Xiang-Yang and Du, Haohua and Chen, Linlin. *2019 IEEE/ACM 27th International Symposium on Quality of Service.*

Conti, M. N. D. a. V. L., 2016. A survey of man in the middle attacks.. *IEEE Communications Surveys & Tutorials 18.3 .*

Couretas, J. M., 2022. Cyber Security and Defense for Analysis and Targeting. *In An Introduction to Cyber Analysis and Targeting,.*

David, 2017. [Online]   
Available at: https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861

Demertzis, K. P. K. N. T. S. L. S. a. L. I., 2018. he next generation cognitive security operations center: network flow forensics using cybersecurity intelligence.. *Big Data and Cognitive Computing 2, no. 4.*

Demertzis, K. P. K. N. T. S. L. S. a. L. I., 2018. The next generation cognitive security operations center: network flow forensics using cybersecurity intelligence.. *Big Data and Cognitive Computing 2.*

Demirezen, M. U., 2021. Image Based Malware Classification with Multimodal Deep Learning. *International Journal of Information Security Science 10, no. 2.*

Ding, J. C. L. a. B. L., 2022. A Data-Driven Based Security Situational Awareness Framework for Power Systems.. *Journal of Signal Processing Systems.*

dmin, 2022. [Online]   
Available at: https://www.simplilearn.com/tutorials/machine-learning-tutorial/naive-bayes-classifier

Duna, Y. T. M. F. A. R. M. F. Z. T. F. B. a. A. F., 2021. Grasp on Next Generation Security Operation Centre (NGSOC): Comparative Study.. *Int. J. Nonlinear Anal. Appl 12, no. 2.*

Dwivedi, S. M. V. a. S. T., 2022. Defense against distributed DoS attack detection by using intelligent evolutionary algorithm. *International Journal of Computers and Applications 44.3 .*

economics, P. m. f. s. o., 2006. Brian Monahan. *economic of securing network.*

embre, 2020. [Online]   
Available at: https://www.kaggle.com/datasets/trinhvanquynh/ember-for-static-malware-analysis

Emmanuel, T. T. M. D. M. T. S. B. M. a. O. T., 2021. A survey on missing data in machine learning.. *ournal of Big Data 8, no. 1.*

Eskelinen, T., 2022. Development of Open-Source SIEM and Security Operation Centre in A Company.

Faulkner, B., 2007. Hacking into data breach notification laws.. *Fla. L. Rev. 59.*

Gibert Llauradó, D. J. P. C. C. M. P. a. Q. L., 2022. Fusing feature engineering and deep learning: A case study for malware classification. *Expert Systems with Applications, 2022, vol. 207.*

Gilda, S., 2017. Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection.. *In 2017 IEEE 15th student conference on research and development (SCOReD).*

Gomes, C. Z. J. a. H. Y., 2021. Insurance fraud detection with unsupervised deep learning.. *ournal of Risk and Insurance 88.3.*

Gupta, A. a. M. C. L., 2022. Comparative Analysis of Numerous Approaches in Machine Learning to Predict Financial Fraud in Big Data Framework.. *Soft Computing: Theories and Applications. Springer, Singapore, 2022. 107-123..*

H. Jiang, Z. H. G. Y. a. H. Z., 2020. ‘‘Network intrusion detection based on PSO-xgboost model,. *IEEE.*

Hanafy, M. a. R. M., 2022. Classification of the Insureds Using Integrated Machine Learning Algorithms: A Comparative Study.. *Applied Artificial Intelligence .*

Haneesh, I. M. S. a. K. N. S., 2018. A Study on Insurance Amount Prediction from Classifying the Cars.. *IMPACT: International Journal of Research in Engineering & Technology 9, no. 2: 31-36..*

Han, N. J. W. X. F. J. W. S. Z. S. X. a. X. L., 2020. Transferable linear discriminant analysis.. *IEEE Transactions on Neural Networks and Learning Systems 31, no. 12 .*

Hansson, A. a. H. C., 2022. Insurance Fraud Detection using Unsupervised Sequential Anomaly Detection..

Han, W. J. X. Y. W. L. H. Z. K. a. L. M., 2019. MalDAE: Detecting and explaining malware based on correlation and fusion of static and dynamic characteristics. *Computers & Security 83 .*

Happa, J. I. A. M. H. T. B.-R. M. G. a. S. C., 2021. Assessing a decision support tool for soc analysts.. *Digital Threats: Research and Practice.*

Harrison, C. J. a. C. J. S.-G., 2021. Machine learning in medicine: a practical introduction to natural language processing.. *BMC Medical Research Methodology 21, no. 1.*

Haslum, K. A. A. a. S. K., 2007. Dips: A framework for distributed intrusion prediction and prevention using hidden markov models and online fuzzy risk assessment.. *In Third International Symposium on Information Assurance and Security, pp. 183-190. IEEE.*

Huang, Z. Q. W. Y. C. a. X. J., 2020. A survey on machine learning against hardware trojan attacks: Recent advances and challenges. *IEEE Access 8 .*

Idika, N. a. A. P. M., 2007. A survey of malware detection techniques.. *Purdue University 48.2.*

Itri, B. M. Y. M. Q. &. O. B., 2019. Performance comparative study of machine learning algorithms for automobile insurance fraud detection.. *In 2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS) (pp. 1-4). IEEE..*

Itri, B. Y. M. Q. M. a. B. O., 2019. Performance comparative study of machine learning algorithms for automobile insurance fraud detection. *In 2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS.*

Itri, B. Y. M. Q. M. a. B. O., 2019. Performance comparative study of machine learning algorithms for automobile insurance fraud detection.. *In 2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS), pp. 1-4. IEEE.*

Jacobs, P. A. A. a. B. I., 2013. Classification of security operation centers.. *In Information Security for South Africa, pp. 1-7. IEEE.*

Jamalpur, S. et al., 2018. Dynamic Malware Analysis Using Cuckoo Sandbox. *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT).*

jason, 2016. [Online]   
Available at: https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

Jindal, N., 2020. Automated event prioritization for security operation center using graph-based features and deep learning. *PhD diss..*

Kathleen, 2021. [Online]   
Available at: https://www.techtarget.com/searchenterpriseai/feature/How-to-build-a-machine-learning-model-in-7-steps

Ketu, S. a. P. K. M., 2020. Performance analysis of machine learning algorithms for iot-based human activity recognition.. *In Advances in electrical and computer technologies, pp. 579-591. Springer, Singapore,.*

Khaleghi, M. M. R. A. a. M. R., 2022. Comprehensive Comparison of Security Measurement Models. *Journal of Applied Security Research.*

Kokulu, F. B. A. S. T. B. Y. S. Z. Z. A. D. a. G.-J. A., 2019. Matched and mismatched socs: A qualitative study on security operations center issues.. *In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security.*

Ko, N. Y. S. H. R. A. W. a. G. S. C., 2020. Association of insurance status and racial disparities with the detection of early-stage breast cancer.. *JAMA oncology 6, no. 3 (2020): 385-392..*

Kumar, S. B. P. S. a. V. K., 2021. A Semantic Machine Learning Algorithm for Cyber Threat Detection and Monitoring Security.. *In 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), pp. 1963-1967. IEEE,.*

Kwak, M.-J. H. G. K. K. C. S.-K. K. J. Y. S. J. L. P. A. L. e. a., 2018. Rhizosphere microbiome structure alters to enable wilt resistance in tomato.. *Nature biotechnology 36, no1.*

Lewis, J. A., 2022. Assessing the risks of cyber terrorism, cyber war and other cyber threats.. *Washington, DC: Center for Strategic & International Studies,.*

Li, N. M. S. a. Y. G., 2020. A systematic review of unsupervised learning techniques for software defect prediction.. *Information and Software Technology 122 (2020): 106287..*

Liu, D. Z. X. Y. Z. a. C. F., 2019. Heat map visualisation of fire incidents based on transformed sigmoid risk model.. *Fire Safety Journal 109.*

Liu, S. P. F. S. W. K. S. a. J. C., 2022. Enhancing Malware Analysis Sandboxes with Emulated User Behavior.. *Computers & Security.*

ltex, 2021. [Online]   
Available at: https://www.altexsoft.com/blog/datascience/preparing-your-dataset-for-machine-learning-8-basic-techniques-that-make-your-data-better/

Luque, A. A. C. A. M. a. A. d. L. H., 2019. The impact of class imbalance in classification performance metrics based on the binary confusion matrix.. *Pattern Recognition 91.*

Mahamadu, A.-M. L. M. a. C. B., 2013. Challenges to BIM-cloud integration: Implication of security issues on secure collaboration.. *2013 IEEE 5th International Conference on Cloud Computing Technology and Science. Vol. 2. IEEE.*

Majeed, A. F. A. M. A. a. N. J., 2019. Near-miss situation based visual analysis of SIEM rules for real time network security monitoring.. *Journal of Ambient Intelligence and Humanized Computing 10, no. 4.*

Majid, A.-A. M. A. J. A. E. K. a. A. S., 2021. A review of artificial intelligence based malware detection using deep learning. *Materials Today: Proceedings.*

Mallik, A., 2019. Man-in-the-middle-attack: Understanding in simple words.. *Cyberspace: Jurnal Pendidikan Teknologi Informasi 2.2.*

Marcus, D. J., 2018. The Data Breach Dilemma: Proactive Solutions for Protecting Consumers'. *Personal Information." Duke LJ 68 .*

McLeod, A. a. D. D., 2022. Information security policy non-compliance: Can capitulation theory explain user behaviors?. *Computers & Security 112 (2022): .*

Mijwil, M. M., 2020. Malware Detection in Android OS Using Machine Learning Techniques.. *Data Science and Applications 3, no. 2 .*

Mohanta, A. a. A. S., 2020. Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware. *APRESS.*

Mohmand, M. I. H. H. A. A. K. U. U. M. Z. A. A. M. R. I. U. R. a. M. H., 2022. A Machine Learning-Based Classification and Prediction Technique for DDoS Attacks. *IEEE.*

Monnappa, K., 2015. Automating linux malware analysis using limon sandbox.. *Black Hat Europe.*

Nazareno, B., 2021. Dynamic Malware Analysis: Contrast between Physical and Virtual Environment. *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education.*

Nian, K. H. Z. A. T. T. C. a. Y. L., 2016. Auto insurance fraud detection using unsupervised spectral ranking for anomaly.. *The Journal of Finance and Data Science 2, no. 1 .*

nist, 2021. Issue https://www.itl.nist.gov/div898/handbook/eda/section1/eda11.htm.

Oh, C. J. H. a. H. R., 2021. A Survey on TLS-Encrypted Malware Network Traffic Analysis Applicable to Security Operations Centers.. *Applied Sciences 12, no. 1.*

onel, 2018. [Online]   
Available at: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

Onwubiko, C., 2015. Cyber security operations centre: Security monitoring for protecting business and supporting cyber defense strategy.. *2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA). .*

Oprea, A. Z. L. R. N. a. K. B., 2018. Made: Security analytics for enterprise threat detection.. *In Proceedings of the 34th Annual Computer Security Applications Conference, pp. 124-136. .*

O'Reilly, C. A. G. M. A. I. a. S. R., 2014. Anomaly detection in wireless sensor networks in a non-stationary environment.. *EEE Communications Surveys & Tutorials 16, no. 3.*

Osman, A. I. A. A. N. A. M. F. C. Y. F. H. a. A. E.-S., 2021. Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia. *in Shams Engineering Journal 12, no. 2.*

Pranavi, P. S. H. D. S. S. S. K. S. K. a. B. H. S., 2020. Analysis of Vehicle Insurance Data to Detect Fraud using Machine Learning.. *International Journal for Research in Applied Science & Engineering Technology (IJRASET) 8, no. 7.*

Pranavi, P. S. S. H. D. K. S. S. K. S. &. S. B. H., 2020. Analysis of Vehicle Insurance Data to Detect Fraud using Machine Learning.. *International Journal for Research in Applied Science & Engineering Technology (IJRASET), 8(7).*

priyakun, 2021. [Online]   
Available at: https://www.knowledgehut.com/blog/data-science/linear-discriminant-analysis-for-machine-learning

pyware, n.d. [Online]   
Available at: https://us.norton.com/internetsecurity-malware-spyware.html

Qu, Y. B. F. W. Z. R. T. M. N. H. G. Y. Y. a. X. H., 2018. Product-based neural networks for user response prediction over multi-field categorical data.. *ACM Transactions on Information Systems (TOIS) 37, no. 1.*

Rajkomar, A. J. D. a. I. K., 2019. Machine learning in medicine.. *New England Journal of Medicine 380.14 .*

ransomware, n.d. [Online]   
Available at: https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware.html

Rashidi, B. C. F. a. E. B., 2017. Android resource usage risk assessment using hidden Markov model and online learning.. *Computers & Security 65.*

Rathore, H. S. A. S. K. S. a. M. S., 2018. Malware detection using machine learning and deep learning. *In International Conference on Big Data Analytics, pp. 402-411. Springer, Cham.*

Ravi, A. a. V. C., 2022. Static Malware Analysis using ELF features for Linux based IoT devices. *2022 35th International Conference on VLSI Design and 2022 21st International Conference on Embedded Systems (VLSID). IEEE.*

Resul, D. A. S. a. M. Z. G., 2020. Analysis of cyber-attacks in IoT-based critical infrastructures. *International Journal of Information Security Science 8.4.*

Rodríguez, P. M. A. B. J. G. a. S. E., 2018. Beyond one-hot encoding: Lower dimensional target embedding.. *Image and Vision Computing 75.*

Rodríguez, P. M. A. B. J. G. a. S. E., 2018. Beyond one-hot encoding: Lower dimensional target embedding.". *mage and Vision Computing 75 (2018): 21-31..*

Roy, R. a. K. T. G., 2017. Detecting insurance claims fraud using machine learning techniques.. *In 2017 international conference on circuit, power and computing technologies (ICCPCT), .*

Saad, S. W. B. a. H. E., 2019. The curious case of machine learning in malware detection. *arXiv preprint arXiv:1905.07573.*

Sagi, O. a. L. R., 2018. Ensemble learning: A survey.. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8, no. 4 .*

Salah, A. T. M. A. H. M. I. A. Y. H. S. Z. M. E. a. G. A. K., 2022. Android Static Malware Detection using tree-based machine learning approaches. *In 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), pp. 3-10.*

Sanatan, 2019. [Online]   
Available at: https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eeecb78b422a

Schrider, D. R. a. A. D. K., 2019. Supervised machine learning for population genetics: a new paradigm. *Trends in Genetics 34, no. 4.*

Sendi, A. S. M. D. M. J. a. M. C., 2012. Real time intrusion prediction based on optimized alerts with hidden markov model.. *Journal of networks 7, no. 2 .*

Serpanos, D. P. M. G. X. a. V. I., 2021. Sisyfos: A modular and extendable open malware analysis platform. *Applied Sciences 11, no. 7 .*

Shahjee, D. a. N. W., 2022. "Integrated Network and Security Operation Center: A Systematic Analysis (February 2022).. *IEEE.*

Sheshasaayee, A. a. S. S. T., 2019. A purview of the impact of supervised learning methodologies on health insurance fraud detection.. *In Information Systems Design and Intelligent Applications.*

Shipe, M. E. S. A. D. F. F. a. E. L. G., 2019. Developing prediction models for clinical use using logistic regression: an overview.. *Journal of thoracic disease 11, no. Suppl 4 .*

Shmelkin, R. L. W. a. T. F., 2021. Generating Master Faces for Dictionary Attacks with a Network-Assisted Latent Space Evolution.. *In 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), pp. 01-08..*

Singh, P. a. V. R., 2021. Attack and intrusion detection in cloud computing using an ensemble learning approach.. *International Journal of Information Technology 13, no.2.*

Sinha, M. S. G. S. S. R. a. S. D., 2021. Sniffer: A machine learning approach for DoS attack localization in NoC-based SoCs.. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems 11, no. 2.*

Slay, N. M. &. J., 2016. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective, 25:1–3, 18–31.*

Sopan, A. M. B. M. M. a. R. K., 2019. Machine Learning for Security Operation Center..

statistic, 2022. *malware-attacks-per-year-worldwide.* [Online]   
Available at: https://www.statista.com/statistics/873097/malware-attacks-per-year-worldwide/

Stojmenovic, I. a. S. W., 2014. The fog computing paradigm: Scenarios and security issues.. *In 2014 federated conference on computer science and information systems, pp. 1-8. IEEE.*

sunil, 2017. [Online]   
Available at: https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Sur, P. a. E. J. C., 2019. A modern maximum-likelihood theory for high-dimensional logistic regression.. *Sur, Pragya, and Emmanuel J. Candès. "A modern maximum-likelihood theory for high-dimensional logistic regression." Proceedings of the National Academy of Sciences 116, no. 29.*

Susukailo, V. I. O. a. S. V., 2020. Analysis of the attack vectors used by threat actors during the pandemic.. *In 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), vol. 2, pp. 261-264.*

swastik, 2020. Issue https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/.

Syed, R., 2019. Enterprise reputation threats on social media: A case of data breach framing.. *The Journal of Strategic Information Systems 28,.*

Taheri, L. A. F. A. K. a. A. H. L., 2019. Extensible android malware detection and family classification using network-flows and API-calls.. *International Carnahan Conference on Security Technology (ICCST). .*

Tama, B. A. a. K. H. R., 2015. Performance analysis of multiple classifier system in DoS attack detection. *Tama, Bayu Adhi, and Kyung Hyune Rhee. "Performance analysis of multiple classifier system in DoS attack detection." International Workshop on Information Security Applications. Springer, Cham.*

Team, C., 2022. *Random forest.* [Online]   
Available at: https://corporatefinanceinstitute.com/resources/data-science/random-forest/

Tertytchny, G. N. N. a. M. K. M., 2020. Classifying network abnormalities into faults and attacks in IoT-based cyber physical systems using machine learning. *Microprocessors and Microsystems 77.*

Threats, U. a. P. M. t. M. C. S., 2015. Venkatesh Jaganathan ,1 Priyesh Cherurveettil ,1 and Premapriya Muthu Sivashanmugam1. *The Scientific World Journal, vol. .*

Tianqi Chen, T. H., 2017. xgboost: eXtreme Gradient Boosting.

tony, 2019. [Online]   
Available at: https://towardsdatascience.com/understanding-random-forest-58381e0602d2

Vyas, S. a. S. S., 2022. Fraud Detection in Insurance Claim System: A Review.. *In 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), pp. 922-927.*

worm, n.d. [Online]   
Available at: https://en.wikipedia.org/wiki/Worm

Wu, Y. a. K. H., 2018. Group normalization. *In Proceedings of the European conference on computer vision (ECCV), pp. 3-19.*

Xiao, M. C. G. G. S. Y. C. a. C. J., 2021. "Image-based malware classification using section distribution information.. *Computers & Security 110`.*

Yoshioka, K. Y. H. T. O. a. T. M., 2017. Your sandbox is blinded: Impact of decoy injection to public malware analysis systems. *Information and Media Technologies 6, no. 2.*

Yuhua Yin, J. J.-J. W. X. A. S. J. Z. F. S. J. K., 2022. A Hybrid Feature Selection Method for MLP-based Network Intrusion Detection on UNSW-NB15 Dataset. *Cryptography and Security (cs.CR).*

Zeeshan, M. Q. R. M. A. B. M. K. S. H. J. S. A. H. a. A. R., 2021. Protocol-Based Deep Intrusion Detection for DoS and DDoS Attacks Using UNSW-NB15 and Bot-IoT Data-Sets.. *IEEE Access 10.*

Zhang, J. Y.-T. J. a. X. L., 2019. "Cross-site scripting (XSS) detection integrating evidences in multiple stages. *In Proceedings of the 52nd Hawaii International Conference on System Sciences..*

Zhou, J. H. M. K. Y. &. I. A., 2020. Evaluation to Classify Ransomware Variants based on Correlations between APIs. *In ICISSP (pp. 465-472)..*