



# UNIVERSITY OF ALBERTA

Master of Science in Internetworking (MINT)  
Department of Computing Science

MINT 709 Capstone Project Report  
On  
Analysis of Artificial Intelligence Techniques to Detect,  
Prevent, Analyze and Respond to Malware

By Lydia Pilli

Under the guidance of  
Michael Spaling

Fall 2022

## Acknowledgement

I want to thank my project mentor Michael Spaling and program director, Dr. Mike McGregor, for allowing me to work on this intriguing project.

I would especially like to thank my mentor for his empathy, support and help while I navigated through the challenging aspects of my project and tried to make this lighter on me.

I want to thank my friend for his kind help and advice.

I thank my family and friends who have encouraged and supported me through this.

Moreover, lastly, I thank God for giving me the ability and wisdom to work on this project.

# Table of Contents

<b>ABSTRACT .....</b>	<b>1</b>
<b>CHAPTER 1: MALWARE .....</b>	<b>2</b>
1.1 WHAT IS MALWARE? .....	3
1.2 HISTORY OF MALWARE: A BRIEF TIMELINE.....	3
1.3 ATTACK VECTORS: .....	8
1.3.1 <i>Passive attack vectors</i> .....	8
1.3.2 <i>Active attack vectors</i> .....	9
1.4 PROBLEMS INTRODUCED BY MALWARE:.....	13
1.5 COMMON TYPES OF MALWARE .....	15
1.6 IBM’S DEEPLCKER: AN AI-POWERED MALWARE .....	18
<b>CHAPTER 2: ARTIFICIAL INTELLIGENCE .....</b>	<b>20</b>
2.1 WHAT IS ARTIFICIAL INTELLIGENCE?.....	21
2.2 HISTORY OF ARTIFICIAL INTELLIGENCE.....	21
2.3 TYPES OF ARTIFICIAL INTELLIGENCE .....	24
2.4 FIELDS OF AI.....	25
2.5 MACHINE LEARNING .....	27
2.6 TYPES OF MACHINE LEARNING:.....	28
2.6.1 <i>Supervised Machine Learning</i> .....	28
2.6.2 <i>Unsupervised Machine Learning</i> .....	33
2.6.3 <i>Reinforced Learning</i> .....	34
2.7 DEEP LEARNING/ NEURAL NETWORKS IN MACHINE LEARNING .....	35
2.7.1 <i>Neural Network</i> .....	35
2.7.2 <i>Convolutional Neural Network</i> .....	37

2.7.3 Deep Learning .....	37
2.8 CONFUSION MATRIX [60] .....	39
2.9 CROSS-VALIDATION TECHNIQUES .....	40
2.10 GENERALIZATION, OVERFITTING, UNDERFITTING .....	42
2.11 ADVANTAGES AND DISADVANTAGES OF AI .....	43
2.11.1 Advantages .....	43
2.11.2 Disadvantages.....	44
<b>CHAPTER 3: DETECT, PREVENT, ANALYZE AND RESPOND .....</b>	<b>45</b>
3.1 DETECT .....	46
3.2 IMPORTANCE OF MALWARE DETECTION ON AN ORGANIZATIONAL LEVEL.....	47
3.3 MALWARE DETECTION TECHNIQUES .....	47
3.4 PREVENT .....	49
3.5 ANALYSE.....	50
3.6 IMPORTANCE OF MALWARE ANALYSIS.....	51
3.7 RESPONSE.....	52
<b>CHAPTER 4: LAB IMPLEMENTATION .....</b>	<b>54</b>
4.1 INTRODUCTION .....	55
4.2 DATASET.....	55
4.3 METHODS.....	57
4.3.1 Data.....	57
4.3.2 Classification Model – Sequential DNN (Deep Neural Network) with Softmax Regression.....	57
4.3.3 Explanation .....	60
4.4. RESULTS AND DISCUSSION.....	61

4.5 LIMITATIONS ..... 67

4.6 KEY FINDINGS: AI IN DETECTION, PREVENTION, ANALYSIS, AND RESPONSE TO MALWARE..... 67

**CHAPTER 5: CONCLUSION & FUTURE CONSIDERATIONS .....68**

FUTURE WORK ..... 69

**WORKS CITED .....70**

# Table of Figures

FIGURE 1: REVETON’S METHOD FOR ASKING FOR RANSOM [6] .....	5
FIGURE 2: CRYPTOLOCKER ASKING FOR RANSOM IN BITCOIN [8] .....	6
FIGURE 3: RANSOM COLLECTED IN 2020 AND 2021, ACCORDING TO SONICWALL [11] .....	7
FIGURE 4: DISTRIBUTED DENIAL OF SERVICE (DDoS) ATTACK [13] .....	9
FIGURE 5: A PHISHING EMAIL. NOTE THE SPELLING MISTAKES “RECIEVED” AND “DISCREPANCY.” [16] .....	11
FIGURE 6: AI-POWERED CONCEALMENT [28] .....	19
FIGURE 7: THE TURING TEST [100] .....	21
FIGURE 8: THE IMITATION GAME [99].....	21
FIGURE 9: A BRIEF TIMELINE OF AI UNTIL THE 2000s [30] .....	22
FIGURE 10: ARTIFICIAL INTELLIGENCE VS. MACHINE LEARNING VS. DEEP LEARNING .....	26
FIGURE 11: TYPES OF MACHINE LEARNING [39]. FIGURE AVAILABLE VIA LICENSE: CREATIVE COMMONS ATTRIBUTION 4.0 INTERNATIONAL .....	28
FIGURE 12: SUPERVISED MACHINE LEARNING .....	29
FIGURE 13: LINEAR REGRESSION [46].....	31
FIGURE 14: LOGISTIC REGRESSION [47] .....	32
FIGURE 15: UNSUPERVISED MACHINE LEARNING: CLUSTERING [50].....	33
FIGURE 16: REINFORCED LEARNING [52] .....	34
FIGURE 17: A BIOLOGICAL NEURON [55].....	35
FIGURE 18: ARTIFICIAL NEURON .....	36
FIGURE 19: A NEURAL NETWORK .....	36
FIGURE 20: CONVOLUTIONAL NEURAL NETWORK [58] .....	37
FIGURE 21: DEEP NEURAL NETWORK .....	38
FIGURE 22: A CONFUSION MATRIX [60] .....	39
FIGURE 23: ELABORATE CONFUSION MATRIX [60] .....	39
FIGURE 24: HOLD-OUT CROSS-VALIDATION [61].....	40
FIGURE 25: K-FOLD CROSS-VALIDATION [61] .....	41

FIGURE 26: UNDERFITTING VS. JUST RIGHT VS. OVERFITTING [62].....	42
FIGURE 27: DATASET CREATION [90].....	56
FIGURE 28: AN ILLUSTRATION OF DNN WITH SOFTMAX OUTPUT LAYER.....	59
FIGURE 29: TRAINING AND VALIDATION ACCURACY OF THE NEURAL NET.....	61
FIGURE 30: TRAINING AND VALIDATION LOSS OF THE NEURAL NET .....	61
FIGURE 31: CONFUSION MATRIX OF IMPLEMENTATION 1.....	62
FIGURE 32: TRAINING AND VALIDATION ACCURACY USING HOLD-OUT CROSS-VALIDATION FOR IMPLEMENTATION 2. .....	65
FIGURE 33: TRAINING AND VALIDATION LOSS USING HOLD-OUT CROSS-VALIDATION FOR IMPLEMENTATION 2.....	65
FIGURE 34: CONFUSION MATRIX FOR IMPLEMENTATION 2 .....	66

## Abstract

Malware has been posing a significant problem to almost every organization. Every organization online and every user using the services of the internet is susceptible to attacks. The preparators of these attacks are humans; malware is the tool they use to exploit the systems. The “malicious” software is designed to carry out activities such as spying on user activity, encrypting critical information, making use of the host system’s applications without the user’s knowledge to access and steal critical data, creating entry and exit points for an attacker to enter and exit the victim’s environment as they please – creating backdoors. Malicious activity can be done using the applications on the system is not done without interacting with the operating system through the application programming interface (API) calls.

Many human resources are expended on monitoring systems to notice any anomalies caused by the running applications. It is a relatively slow process to detect such anomalies when the count of systems to be monitored on an organizational level increase from thousands to tens of thousands. Artificial intelligence (AI), a trendy industry buzzword, can expedite the detection of any malicious activity and assist IT professionals.

This report draws a basic understanding of malware and artificial intelligence concepts. It also presents how artificial intelligence can be used to detect malware. Anything detected can be prevented, analyzed, and responded to. In the lab implementation section, with the help of an AI model, it is demonstrated how artificial intelligence can detect malicious activity by malware through its invoked API calls. A sequential deep-neural network with a multi-class classification algorithm – softmax regression, is developed to classify malware based on its API calls.

Artificial intelligence serves as an excellent tool for detecting malware. AI's learning capabilities to learn from vast amounts of data make it a powerful tool. Especially in the cybersecurity industry, where the possibility of zero-day attacks is never nil, AI could help identify these attacks by learning from all the previous data, recognizing patterns, and predicting the likelihood of one.

This report could serve as a reference to anyone seeking to develop an AI application to detect malware and educate one on malware and AI and AI detection capabilities.



# Chapter 1: Malware

## 1.1 What is Malware?

Malware is short for malicious software designed to exploit the vulnerabilities of an existing system to intrude and steal valuable information or render them unusable, in today's world, mainly with the intent of extorting money from its owner. The attacker looks for security loopholes in anti-malware solutions or configurations of the devices. Upon finding a vulnerability, the attacker tries to intrude **undetected**; this is critical to exploitation. Malware is the tool used to exploit the system.

The term 'Malware' profiles the existence of its various types – Trojan, Adware, Spyware, Ransomware, and Virus.

However, where did it all begin?

## 1.2 History of Malware: A Brief Timeline

The idea of Malware was first formulated in the 1940s by German mathematician Jon von Neuman, and a paper, 'Theory of Self-Reproducing Automata,' was published in 1966 written by him. The report drew comparisons between biological organisms and a machine. It was about possibly having a computer code that behaved similarly to the natural virus – infecting and damaging the host system, replicating itself and spreading [1].

1971  
"The Creeper"

The idea materialized, and Creeper – the first-ever virus- was created in 1971 by an engineer Bob Thomas. It was a virus created without malicious intent, unlike today. It travelled through the ARPANET, making use of the transport layer protocol NCP (Network Control Program – now obsolete) and would display the following message: [2]

```
"I am the creeper, catch me if you can [2]!"
```

1982  
"Elk Cloner"

The first computer virus, created by a 15-year-old, was designed for and targeted at Apple II users named the "Elk Cloner." It is a boot sector virus. Relatively simple in functioning. It would be transmitted through an infected disk. Upon loading for the fiftieth time, it would display the following poem: [2]

```
Elk Cloner: The program with a personality  
It will get on all your disks  
It will infiltrate your chips  
Yes, it is Cloner!  
It will stick to you like glue  
It will modify RAM too  
Send in the Clone [2]!
```

1982  
“Brain”

The usage of the virus in this case scenario is fascinating. “Brain,” a boot sector virus, was developed by two Pakistani brothers who wanted to proactively discourage the use of pirated copies of their medical software. The virus prevents the machine from booting when someone tries to use a copy of their software. It displays a message/ word of warning, including the contact information of the Pakistani brothers, to encourage the user to contact them to get a legal copy of their software. This virus needed human interaction to spread. Although not dangerous, it spread quickly through Europe and North America via infected disks. This virus marked the alteration of the cybersecurity world into the form we know today. Until now, human involvement has been required to spread the virus. [2]

The message was as follows:

```
Welcome to the Dungeon (c) 1986 Amjads (pvt) Ltd VIRUS_SHOE
RECORD V9.0 Dedicated to the dynamic memories of millions
of viruses who are no longer with us today - Thanks
GOODNESS!!! BEWARE OF THE er ..VIRUS : this program is
catching program follows after these ...$#@%$@!!
Welcome to the Dungeon © 1986 Basit & Amjads (pvt). BRAIN
COMPUTER SERVICES 730 NIZAM BLOCK ALLAMA IQBAL TOWN LAHORE-
PAKISTAN PHONE: 430791,443248,280530. Beware of this
VIRUS... Contact us for vaccination [3]...
```

1988  
“The Morris  
worm”

Then came 1988, and the Adam of worms – Robert Morris developed the Morris worm to test if malware could replicate itself without human involvement. This worm was not malicious but a successful experiment that caused a Denial of Service - DOS attack on the network [2].

DOS attack was caused by exploiting vulnerabilities in UNIX send mail, finger, and rsh/rexec and guessing weak passwords. Within 24 hours, 60,000 (10%) computers connected to the Advanced Research Projects Agency Network (ARPANET) were hit [4] .

1989  
“AIDS Trojan”

The AIDS Trojan – Dr. Joseph Pop developed the world’s first ransomware. He sent out about 20,000 floppy disks worldwide in physical mail. When loaded, it had a questionnaire on the topic “AIDS.” It would encrypt the file names. A ransom of \$189/year (\$466 in 2023) or \$385 (\$948 in 2023) for a lifetime was demanded. The money was not exchanged in digital transactions but in specified physical forms (Money orders, cashier’s checks) sent to the PO Box address in Panama [2].

2000  
"I LOVE YOU  
"

Fast-forwarding to the year 2000 when the I LOVE YOU worm was introduced. An email attachment of the I LOVE YOU worm was sent under the pretext of a love letter. People are generally curious/excited about receiving one; the natural next step would be to open it. Upon opening it, the worm would corrupt files on hard disks and send these love letters to the victim's contact list. It has reportedly infected about 2.5 million PC this way. This attack changed how internet security was perceived until that point [5]. The loss to businesses has been significant.

2003 -  
2012

The cybercrime industry started taking shape from here, and attacks have been advancing into spyware, botnet, and Backdoor. However, so far, it was only created mischief, although it did cause infrastructure damage. Large-scale and disastrous ransomware attacks started taking place around 2011/2012, starting from Reveton, considered the ancestor of all modern ransomware. Reveton's method of demanding ransom is still used in a similar format, by creating a display page explaining what happened, instructions on how to get the decryption key for the encrypted data and how much ransom to pay. The method of demanding ransom now is a derivative of Reveton's [2].



Figure 1: Reveton's method for asking for ransom [6]

2013  
“Cryptolocker”

By 2013 cryptocurrency emerged— a technique to buy and sell using digital currency. During its year of inception, the value of bitcoin was around \$0 [7]. However, as and when the value of bitcoin kept accumulating, it was an opportunity for cybercriminals to demand ransom in crypto. Crypto transactions occur between the buyer and the seller without third-party involvement, primarily banks. Moreover, transactions like this provided more anonymity and ease to the criminal as they would be difficult to trace, and there is no regulatory authority. It could not be more convenient.

CryptoLocker is the first ransomware that demanded ransom in crypto [2].



Figure 2: CryptoLocker asking for ransom in bitcoin [8]

CryptoLocker at least accumulated a ransom of 3 million dollars to the attacker. According to CNE [9].

2019  
“Sodinokibi (REvil)”

A single company called Travelex – a foreign currency exchange company, was infiltrated and attacked by Sodinokibi (a ransomware variant), stealing sensitive customer data (including credit card numbers and insurance details) worth 5G. They (it was a ransomware group) demanded a ransom of \$6 million. After negotiation, the company paid a ransom of \$2.3 million [9].

2021  
“Phoenix  
Locker”

The seventh largest commercial insurer in the USA, CNA Financial, was a victim of the ransomware – Phoenix Locker. It paid \$40 million in ransom to regain its data [9].

**Ransomware as a service** started to rise where the ransomware developers lend or sell their ransomware to lenders who want to exploit a system. They lent it for a fee and sometimes an agreed-upon profit share. The ransomware creators are known as operators and the lenders as affiliates [10].

With the advent of bitcoin and ransomware as a service – ransomware attacks grew exponentially. The amount of ransom being demanded and paid was extravagant. With this, our threat landscape has evolved to what it is today, and cybersecurity has never been more essential to businesses. Other malware attacks were carried out to create mischief during those years, but the surge in ransomware attacks diminished the others.

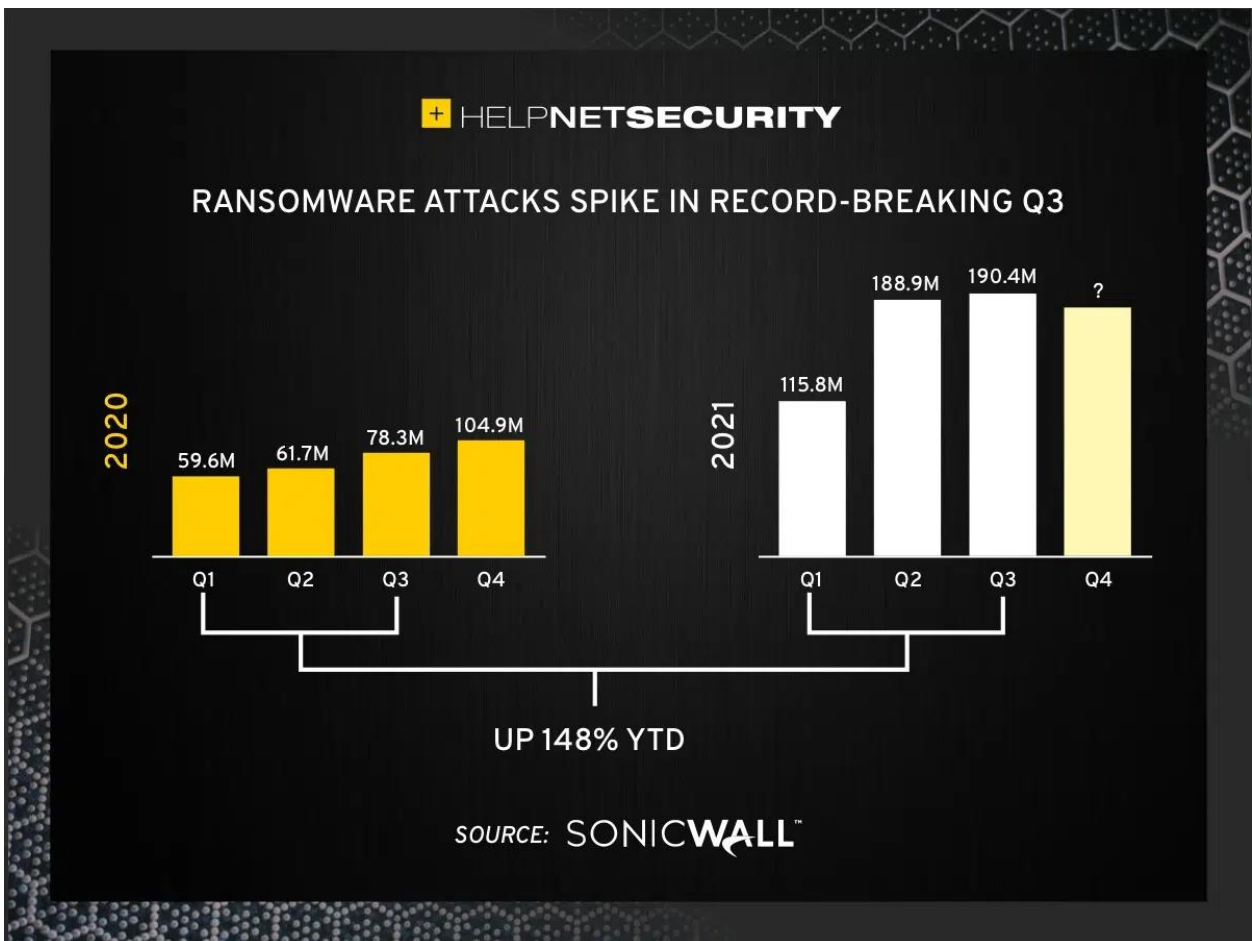


Figure 3: Ransom collected in 2020 and 2021, according to Sonicwall [11]

## 1.3 Attack Vectors:

Attack vectors are methods or techniques cybercriminals use to launch weapons in their arsenal (malware) to gain unauthorized access to cause a data breach in an organization to access and steal invaluable, confidential data. Attack vectors can be categorized as:

### 1.3.1 Passive attack vectors

Attack vectors of this type do not cause any significant damage to the operations of an organization. The attacker observes and gathers or steals information on a potential target. Passive attack vectors include: [12]

- **Traffic analysis**  
An attacker can use network analyzing tools to analyze the traffic going to and from the target system to read or snoop the data between the systems.
- **Eavesdropping**  
When an attacker intercepts any communication channel like phone calls or unencrypted text messages to listen in on the conversations exchanged between the targets, without interrupting the ongoing conversation but as an invisible third party to gather any sensitive information on the target(s).
- **Footprinting**  
In this method, the target company's IP address, domain name, hardware, software, and network infrastructure details are gathered.
- **Spying**  
An attacker tries to stealthily install a device within the target's environment to collect all traffic or data.
- **Wardriving**  
In this attack, the attacker may scan a surrounding or targeted area for any vulnerable WIFI networks to connect to launch an attack. The scanning may be done through a portable antenna.
- **Dumpster diving**  
As the name implies, the attacker goes through the dump or trash can to look for valuable information like credit card bills, receipts, and written passwords.

### 1.3.2 Active attack vectors

These attacks directly interact with the environment. Active attack vectors, when successfully launched, can be cataclysmic, bringing the operations of the organization to an abrupt halt by encrypting data and holding it for a ransom (Ransomware) and/or exploiting the nature of network protocols to overwhelm the servers with requests (DoS), exploiting undiscovered system vulnerabilities to launch an attack (zero-day attack), introducing destructive malware. It can bring a whole organization down to its knees.

- Denial of Service attacks (DoS) and Distributed Denial-of-Service (DDoS) Attacks

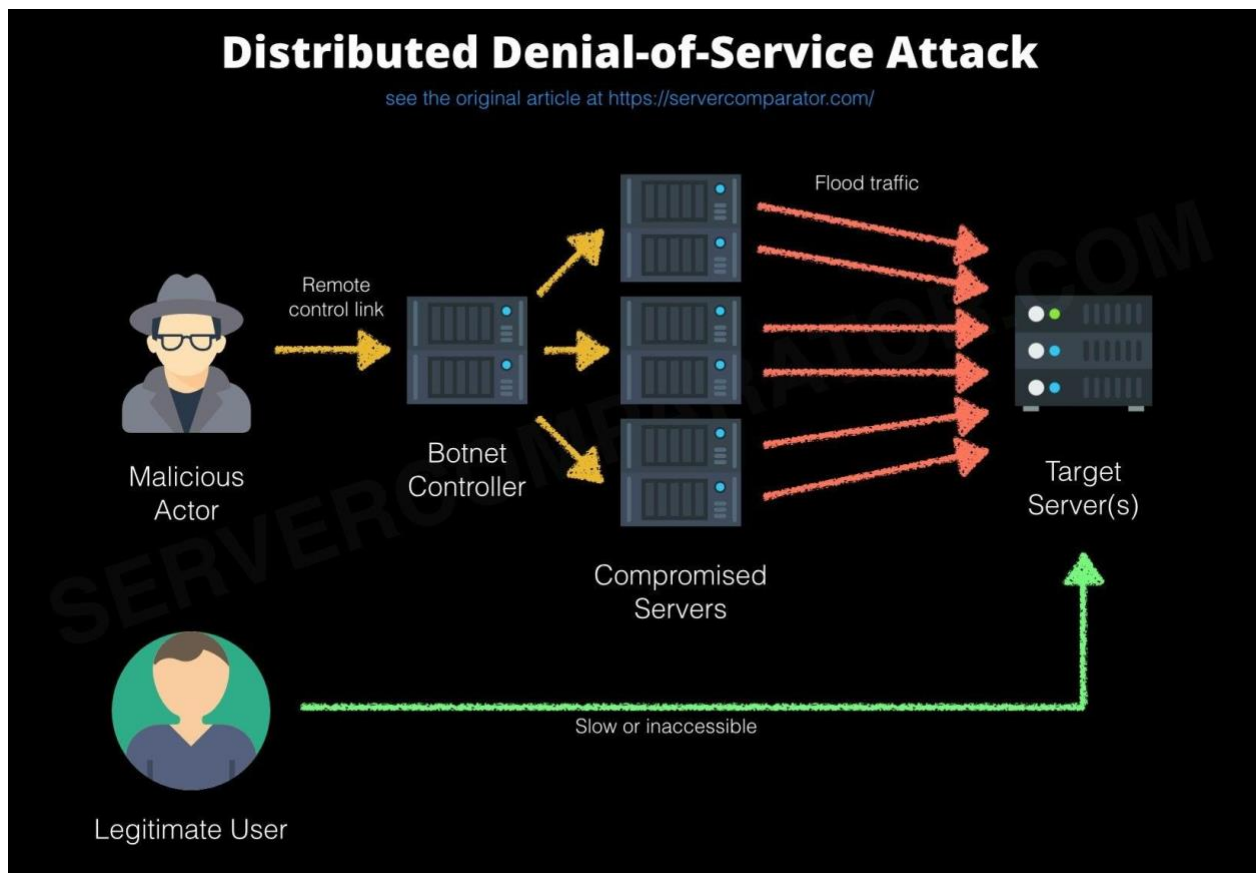


Figure 4: Distributed Denial of Service (DDoS) Attack [13]

This attack is carried out by the malicious user not to steal any information but to overwhelm the targeted organizations' servers with traffic to make them unresponsive to their intended or legitimate users. DoS attack halts the business of the corporate, costing them time and money. The targets are usually high-profile organizations. The Denial-of-Service attack exploits the existing system or network protocol vulnerabilities to crash the system or overwhelms it with packets beyond its processing capabilities, resulting in a buffer overflow [14].



Distributed Denial of Service of Attack occurs when multiple systems send packets simultaneously. Unlike in DoS, where attacks originate from a single location or system, attacks originate from multiple locations [14].

- **Exploiting Software Vulnerabilities**

As aforementioned, there are unknown vulnerabilities existing in any and every system. Software, internet protocols, or organizations are flawed. If the attacker discovers these vulnerabilities before the developer, they can be exploited, and an attack can be launched. This is a zero-day attack since there are zero days to prepare for it.

- **Brute Force attack**

Attackers use brute force attacks (launched using tools that test different words, and alphanumeric combinations, to guess the password) to steal credentials that are not guarded using robust encryption techniques. Using dictionary words as passwords makes it easier for them to force their way in. If they can crack into one of the accounts and if the same password is used elsewhere, rest assured that those accounts are in danger.

- **Phishing**

This is a viral and successful attack vector. In phishing, the attacker sends an alarming email - making it seem very legitimate and sounding as though immediate action must be taken, or else something terrible may happen; they include a link and/or attachment to click on, taking them to a seemingly legitimate website. Let us consider it to be a website impersonating the bank website. If the user responds, valuable information like user credentials will be stolen, and now the attacker has access to the banking data of the victim. Also, just by clicking the link, unaware malware is installed in the system. This unintended download of malicious files without the user's consent is known as **drive-by-download** [15]. Phishing is not necessarily propagated via email but can be sent as an innocent text message and or a message circulated via social media.



Dear valued customer of TrustedBank,

We have recieved notice that you have recently attempted to withdraw the following amount from your checking account while in another country: \$135.25.

If this information is not correct, someone unknown may have access to your account. As a safety measure, please visit our website via the link below to verify your personal information:

<http://www.trustedbank.com/general/custverifyinfo.asp>

Once you have done this, our fraud department will work to resolve this discrepancy. We are happy you have chosen us to do business with.

Thank you,  
TrustedBank

Member FDIC © 2005 TrustedBank, Inc.

Figure 5: A phishing email. Note the spelling mistakes “recieved” and “discrepancy.” [16]

- **Malicious Insider**

Ingenuine or disloyal employees may leak sensitive/confidential information for a price or for other ulterior malicious motives to unauthorized, malicious people (people external to the organization who could potentially harm it for monetary benefit or to propagate political agenda).

- **Misconfigurations**

Attacks can be launched by exploiting misconfigurations on cloud platforms and security devices. An example of misconfiguration would be giving privileged access to standard users [17].

- **Supply Chain Vendors**

Organizations work with multiple other organizations or, in other words, outsource some of their business components. In this chain of trust, the attacker looks for a weak spot and then attacks. It could be the hardware manufacturer or the software-developing organization.

- **Business Email Compromise**

According to Microsoft Security, in this attack, the attacker poses as trusted personnel of the company or maybe impersonates someone from the company itself and gets the victim to disclose confidential information or gets them to pay a fake invoice appearing to be legitimate [18].

## 1.4 Problems Introduced by Malware:

The kind of problems malware introduces depends entirely on the nature of the attack. Monetary loss is one side-effect; on the flip side exist these other problems that Malware raises, according to Kaspersky's encyclopedia:

### **Downtime:**

An immediate consequence of a ransomware attack is that all the network systems are rendered useless, and all files are encrypted. Every system would display an aggravating page demanding ransom for the decryption key when files are accessed. This results in downtime for the attacked organizations until the systems can be restored and run again.

The average downtime after a ransomware attack (focusing on ransomware since it is the most on the rise and its damages shadow every other) is 21 days. Downtime can easily be the costliest aspect of a ransomware attack, considering how the businesses cannot operate during the downtime and the money that goes into the data restoration [19].

Ransom payment does not guarantee that all the data will be restored in most cases. It is in rare cases that ransom payment will restore the complete data. According to Sophos' research (The State of Ransomware 2022), only 4% of organizations that paid the ransom got all their data back, and the average to recover from an attack is reported 4M [20]. During the downtime, other than just recovering all the data from backups, money is invested in restoring the data that has been lost during the data breach. In some cases, data is lost forever, which can be catastrophic (depending on the nature of the data – business-critical, sensitive customer data) for businesses.

### **Data Breach:**

- A data breach occurs through common cyberattacks such as phishing, spyware attack or broken or misconfigured access controls [21].
- An attack is carried out to access data that is an asset or invaluable to the organization.
- During the data breach, cybercriminals will steal sensitive information such as credit card information, banking data, medical data, proprietary organizational information, or trade secrets, which must be guarded with utmost fortification.
- The criminals might use this data to their advantage, such as making illegal purchases on the dark web using credit card information, publishing the organization's secrets, and/or selling this information at a high price to those who could benefit from it. Among the other valuable things in the world, data might as well make it to the top of the list eventually.
- What makes data breaches caused by malware worse is that they can remain **undetected** for a significant time. According to IBM's report (Cost of Data Breach 2022), it takes an average of 277 days (about nine months) to detect and contain a data breach

(207 days to detect and 70 days to contain the breach), and the average total cost of a data breach is \$4.35M [22].

- Data breach results in **data loss** and or **data theft**.

### **Data Loss:**

Data loss refers to data being lost by being deleted, or in the worst-case scenario; data is deleted after being stolen. Data loss can be premeditated or accidental. Accidental damage happens due to natural disasters, hardware and/or software failure, power outages, poorly done backups, human error, data migration, and improper shutdowns. While premeditated data loss occurs when a cybercriminal exploits the vulnerabilities in a system and introduces **malware** to hoard the data and delete it, or in today's world, it is encrypted and held for ransom with no guarantee that all encrypted data can be decrypted, in such cases data is lost forever [23]. Data loss can result from data theft, but they are not the same due to the sources of data loss.

## 1.5 Common types of Malware

With advancements in technology, Malware advanced too. We have evolved from floppy disks to SSDs, World Wide Web 1.0 (static web pages) to World Wide Web 2.0 (dynamic, interactive, e-commerce web pages) with a new possibility of World Wide Web 3.0 (Implementing blockchain technologies – making payments between the buyer and the seller without the involvement of third-party websites. Our networks grew exponentially, with almost 10 billion devices connected to the internet, according to Wired [24]. Malware has evolved from being carried from floppy disk to floppy disk to traversing the network and infiltrating the target system undetected and stealthily, causing mayhem.

All software code could be better. Some loopholes have yet to be discovered, and a zero-day attack is carried out by exploiting these loopholes unknown to the developer and tester. Many tools have been designed which are being used by Cyber Security analysts to detect, prevent, analyze, and respond to malware. These tools are developed to counter the many types of malware in the attacker's arsenal.

Common types of malware are:

- **Fileless Malware**

Fileless Malware, unlike traditional malware, is not in the executable format of file.exe. It is introduced stealthily into the system and uses the infected system's operating system software and applications to achieve the malefactor's goals. Since it works on the backs of legitimate applications, it is hard to detect.

Mode of Operation: [25]

- 1) The attacker exploits a system vulnerability by using web scripting to gain remote access.
- 2) After gaining access, the attacker now tries to steal the credentials to move around the systems with ease.
- 3) Then he proceeds to set up a backdoor.
- 4) Using the system applications now, the attacker exfiltrates the data using FTP.

Example: SamSam Ransomware

- **Spyware**

As the name implies, it spies and accumulates data on the victim's personal or business activity on the infected system- like websites visited, purchases made online, credentials, and payment information all while remaining undercover and undetected until its cover is blown away.

Example: Keylogger

- **Downloader**

Downloader is an application that downloads applications into the user's system without explicitly notifying them. This allows the attacker to download all kinds of malicious applications. It is not malicious but can be used for malicious reasons.

- **Trojan**

Trojan malware is a deceiving malware that appears to be a legitimate application/software concealing its malicious motives until it is time for the attack.

- **Worms**

Worms are self-replicating malware that replicates itself **without human intervention** and spread through the internet. They slow down the system, affecting its operations and overloading networks - sometimes taking them down, consuming bandwidth, stealing data, and opening backdoors [26].

- **Adware**

Adware is more of a nuisance than harmful. This malware installs itself when a link is clicked on (drive-by-download) to visit websites, download freeware, or click on phishing links. One tell-tale sign of adware is that advertisements, not based on browsing activity, start appearing on the visited websites. It may collect information on the victim's interests and sell it to companies for targeted marketing.

Example: Fireball

- **Dropper**

Like Downloader, Dropper is also a type of trojan and helper application. It paves the way for malicious activity by dropping additional malware **that it carries** on the infested system. It is a self-contained malware that drops malicious applications without apparent indicators.

- **Virus**

Like worms, viruses are self-replicating programs that replicate themselves and spread but **with human intervention**.

- **Backdoor**

As the name implies, this is quite analogous to the Backdoor of a regular house. The house owners install this to provide another alternate access into their homes in different scenarios. In the same way, backdoors are installed by software developers or IT engineers to provide alternate access to the system in case a user locks themselves out or diagnoses a technical issue. Cybercriminals can exploit these backdoors to gain admin or root access to the system, or the cybercriminal itself can set up a backdoor to come and go as they please. Backdoors in cybersecurity mean that it is a successful way or approach to gain root access to the system, and it can be authorized or unauthorized (when it is a bad actor), as implied before.

Example: PoisonTap

- **Malicious Bots**

Bots can be used to do some dirty work for cybercriminals – these are known as malicious bots. This software can automate some dirty work cybercriminals do, like stealing data and opening a backdoor. A bot acts as a henchman for the cybercriminal. Upon infestation, it carries out the cybercriminal's goals. A **botnet** is formed when multiple systems are infected with bots; an attacker can now use this botnet to manipulate the systems to do what he desires, for example carrying out a Denial-of-Service attack to bring down a server.

- **Ransomware**

This malware, upon detonating, all data is encrypted and is held for ransom.

Example: Ryuk



## 1.6 IBM's DeepLocker: An AI-Powered Malware

It is very intriguing to see how the malware can make its presence even more scarce than it is already with the help of Artificial Intelligence. IBM developed an AI-powered malware called DeepLocker as a proof-of-concept. DeepLocker gives us a gist of the future of malware attacks, and this information might come in handy to prepare for it.

A little about **Evasive Malware**:

Evasive malware conceals its identity in particular scenarios (as and when in a sandbox) to avoid detection and detonating when its target has been identified or the goal attained or when they are triggered. The example mentioned above scenario is one of the many techniques it employs to avoid detection. In short, it is smart enough to identify its environment and act accordingly to avoid detection. Many techniques are engineered into evasive malware to avoid detection.

Some of the techniques are: [27]

- **Environment Awareness**

As aforementioned, Malware tries to identify the environment it is running in. It keeps itself from executing when it is not in the ideal environment and is in a virtual machine or a sandbox.

- **Timing-based method**

Any malware within a Sandbox is only observed for a specific time. Malware can be tweaked to extend its sleep and prevent itself from being executed (Extended sleep) or schedule its execution (Logic bomb).

- **Stegosplit**

Malware is hidden in images. Images downloaded, especially those available for free, could contain malware – another type of drive-by-download.

**DeepLocker: Ultra-Targeted and Evasive Malware** [28]

DeepLocker malware hides in legitimate applications and usually behaves (does not detonate or obstruct the course of operations of a system) for as long as the target has not been identified. A unique feature of DeepLocker is that it creates its triggers – i.e., locks on a target and detonates upon identification in the real world. It does not need external stimuli like traditional malware. This kind of attack is said to be challenging to reverse engineer mainly since all the “trigger conditions,” which are identical to features of a dataset on which the neural network model is trained, would be too large to list out (“virtually impossible” as stated in the

referenced article). Secondly, uncovering how a model is trained and how they engineer their features/dataset would be a challenge. – proprietary organizational information, recover the malicious payload and target details from the neural net layers. So, the secrets are hidden in the layers of the Neural Net. The concept stated “if this, then that” implies that - if this is the target, then unleash(execute) the malicious code. This concept materializes with the help of a Convolutional Neural Network, one of the many types of Neural Network Models, which will be discussed in Chapter 2 of this paper.



Figure 6: AI-Powered Concealment [28]

IBM’s Security Intelligence tested this concept out successfully by embedding a ransomware-type – “WannaCry,” into a video-conferencing application; this disguised it from detection by the sandbox, Anti-Malware solutions, and malware analysis tools. As their trigger condition, they trained their DNN to detonate upon recognizing a particular face. This person’s face would be the **key** to executing ransomware on the system, and on every system, the application containing ransomware is installed. The key (or trigger condition) could be of various types – a person’s face, a particular organization, probably anything. This could be a stealthy weapon.

# Chapter 2: Artificial Intelligence

## 2.1 What is Artificial Intelligence?

According to John McCarthy, the father of Artificial Intelligence: AI is *“The science and engineering of making intelligent machines, especially intelligent computer programs”* [29].

Artificial Intelligence gives a computer the ability to think.

It is a question of **what a computer can do with the data given to it**, analogous to what a human does with all the data he accumulates in many forms (speech, text, pictures). It combines math, statistics, computer science and enormous data to recreate human intelligence in a computer, known as artificial intelligence. Artificial Intelligence is a derivation of human brain functions that are far more complicated than what we humans have created. There is still much to know about the human brain. We created artificial intelligence concepts based on what little we know of it. All this is to say that, unlike the human brain, artificial intelligence is way less complex.

## 2.2 History of Artificial Intelligence

1950  
“The Turing Test”

AI was a fictional idea used in stories until the concept of “Can machines think?” was first explored realistically by Alan Turing. He was a British polymath who studied this idea's mathematical feasibility through the **Turing Test** in his paper **Computer Machinery and Intelligence**. In the imitation game, an interrogator tries to identify the genders (female or male) of the two people he is talking to (all in different settings). Similarly, in the Turing test, a person tries to tell the difference between a machine and a human by how they respond; if the person cannot tell the difference, it would mean that the computer is artificially intelligent. Though the term “artificial intelligence” was coined only later. The Turing test, among other tests today, is still used to test for AI.

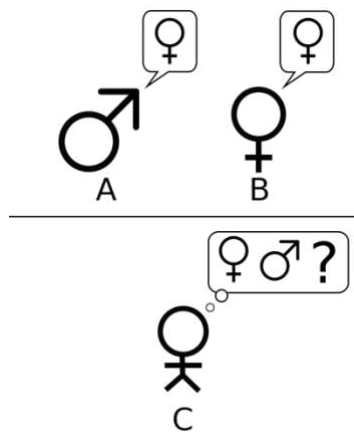


Figure 8: The Imitation game [99]

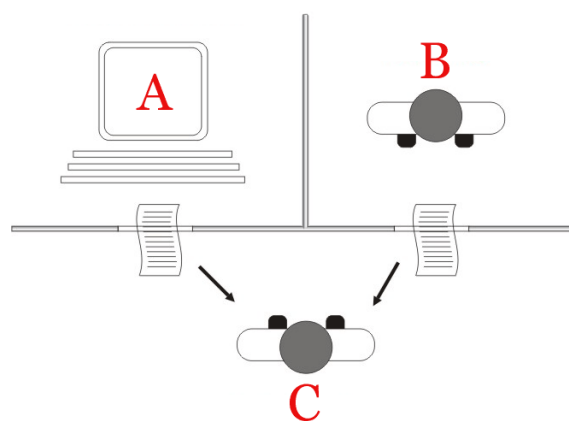


Figure 7: The Turing Test [100]

1956  
"Artificial Intelligence" term was first coined.

Logic Theorist, a program developed by Allen Newell, Cliff Shaw, and Herbert Simon as a proof-of-concept. Problem-solving capabilities of humans were programmed into the application. In 1956, a conference, DSPRAI (Dartmouth Summer Research Project on Artificial Intelligence), was hosted by John McCarthy and Marvin Minsky. It was also where "Artificial Intelligence" was first coined and attributed to McCarthy. While McCarthy gathered researchers and held a discussion on this, Logic Theorist was an application of this concept. Logic Theorist was introduced at the conference. Although this has been a downer for McCarthy, it has been a significant event in the history of AI. The idea that was fiction is now conceivable due to the contribution of the program Logic Theorist [30].

1965  
Moore's Law

Due to computational limitations back then, AI could not be pursued as it is today. Gordon Moore predicted that with every silicon chip, the number of transistors would be doubled each year, implying that computational power would become significantly twice as fast each year. This has been true to a degree [31].

1997  
IBM Deep Blue

IBM Deep Blue is an artificially intelligent computer trained to play chess. It defeated the world chess champion Gray Kasparov. What this meant to the industry is that computers are capable of handling complex problems, and it is a field to venture into further, and this was proved with a game of chess. It was an encouraging invention.

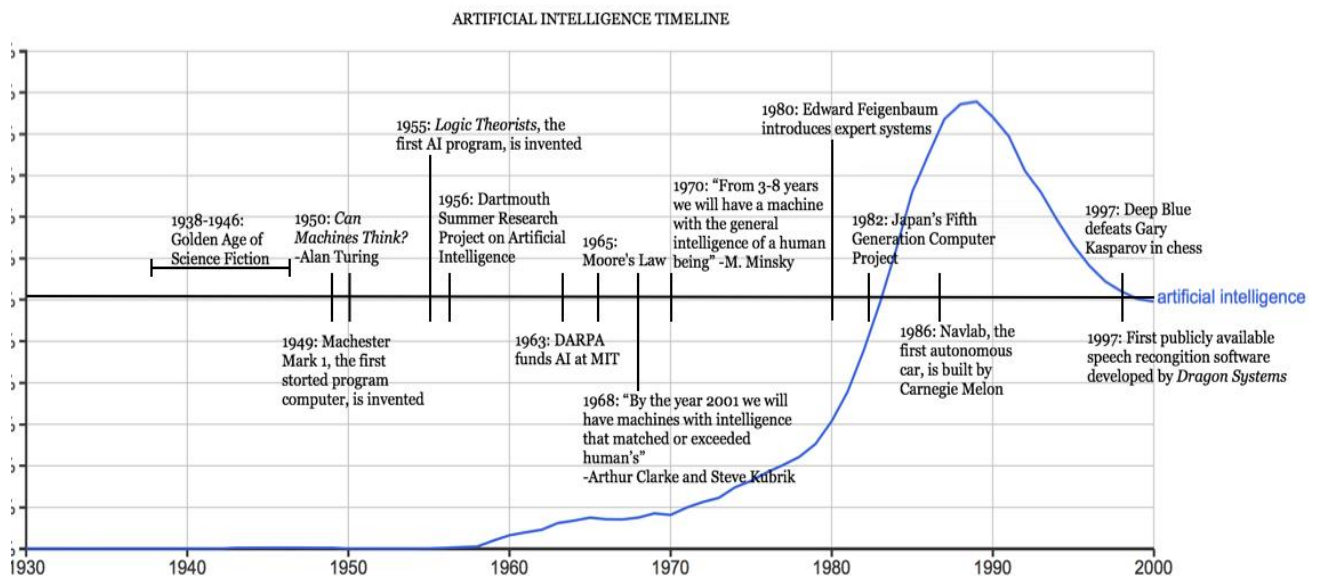


Figure 9: A brief timeline of AI until the 2000s [30]  
The blue line indicates the rise and fall of AI until the year 2000.

### From the 2000s to the present:

- Although AI started a few decades ago, there were fewer applications of AI back then than we have today. This is because to design, run/train and deploy AI models; we need a significant amount of data and technologies to manage this data and the processing power to perform the complex computations required by AI algorithms.
- We have observed significant changes in technology over the years, some of which are significant to AI are **big data**; as years went by, we have amassed much data going up to zettabytes and have invented tools to manipulate and use this data; and the exponential increase in the processing power of computers with the invention of **GPUs** (Graphical processing units) by NVIDIA., and the standard for computational power has increased. GPU was absent a few decades ago.
- Today we have many applications of AI, such as self-driving cars, robotics, and voice assistants, in the medical industry - to diagnose a disease. In business realms: a company can use AI to increase its sales (through prediction models) and in product manufacturing – to identify defects, for example.
- The field of AI is growing by scores. We may have yet to explore its many areas of applications.

## 2.3 Types of Artificial Intelligence

- **ANI - Artificial Narrow Intelligence**

Artificial Narrow intelligence is a branch of AI that deals with a single problem or a single task at a time, hence the term “narrow.” The wide range of applications of AI we see today in the world is ANI. It is the only intelligence we have successfully programmed and implemented. ANI is also known as a weak AI. IBM Deep Blue, Siri, and Netflix recommendations are a few examples of ANI.

- **AGI - Artificial General Intelligence**

AGI is intelligence that a machine has acquired when it responds to external stimuli like a human being. Like humans, it can process, understand, and respond to situations/circumstances. A machine will reach AGI if it has developed cognitive abilities [32]. So far, there is no AGI system.

- **ASI - Artificial Super Intelligence**

ASI means that the intelligence of the system supersedes that of a human. ASI is the AI model that humanity dreads will become true and destroy the world. Nevertheless, the reality is that we are so far off from it. We are at the inception of AI. AGI has yet to be attained, and ASI is still far off.

AGI and ASI are known as strong AI. This report focuses on Artificial Narrow Intelligence (ANI) or weak AI.

## 2.4 Fields of AI

### **Computer Vision:**

Computer vision, as the name implies, gives computers the eyes to see; we train a computer to see a particular object or object and differentiate it from the rest. It is very similar to the human eye, except it is not as powerful as the human eye in the context that the human eye can detect many objects at once – their distance, movement, state, speed, and colours. Computer vision mimics a tiny part of it and can surpass the human eye when imperceptible defects are to be detected. We train the computer using machine learning models by feeding it thousands and thousands of images (For example, we are training a system to detect manufacturing defects in a plate) to identify and differentiate between a good one and a defective piece. By feeding it the image data comprised of images of objects with no defects more in ratio than the one with defects, the model teaches itself to differentiate between the objects, which can be used in the manufacturing industry to look for manufacturing defects [33].

### **Natural Language Processing (NLP):**

Again, like human cognitive abilities, AI is trained to understand language in its speech and text forms with its intent and sentiment. It uses computational linguistics (combines linguistics, computer science and AI to help the computer attain the ability to process human language) [34].

### **Automatic Speech Recognition:**

Automatic Speech Recognition (ASR), or speech recognition, enables a system to convert human audio to text-like transcription.

### **Robotics:**

Robots existed without artificial intelligence. They were just programmed to do repetitive tasks. Machine learning is used in robotics to train robots to see (computer vision), learn, and perform tasks. Robots are trained on large datasets to distinguish between the objects they see and think for themselves for appropriate actions. Various sensors are also integrated into robots, like temperature and humidity sensors, ultrasonic sensors, and vibration sensors, to contribute to its growth in awareness of its surroundings and act likewise [35].



## Machine Learning:

Machine learning is an AI technique. Machine learning, among other tools, makes artificial intelligence possible, or it can be said as machine learning is used to implement AI. Machine learning, computer vision, and NLP are all AI. AI is an umbrella term for the concepts mentioned above. Machine learning uses algorithms and datasets to learn, recognize patterns, and provide insights. To demonstrate better:

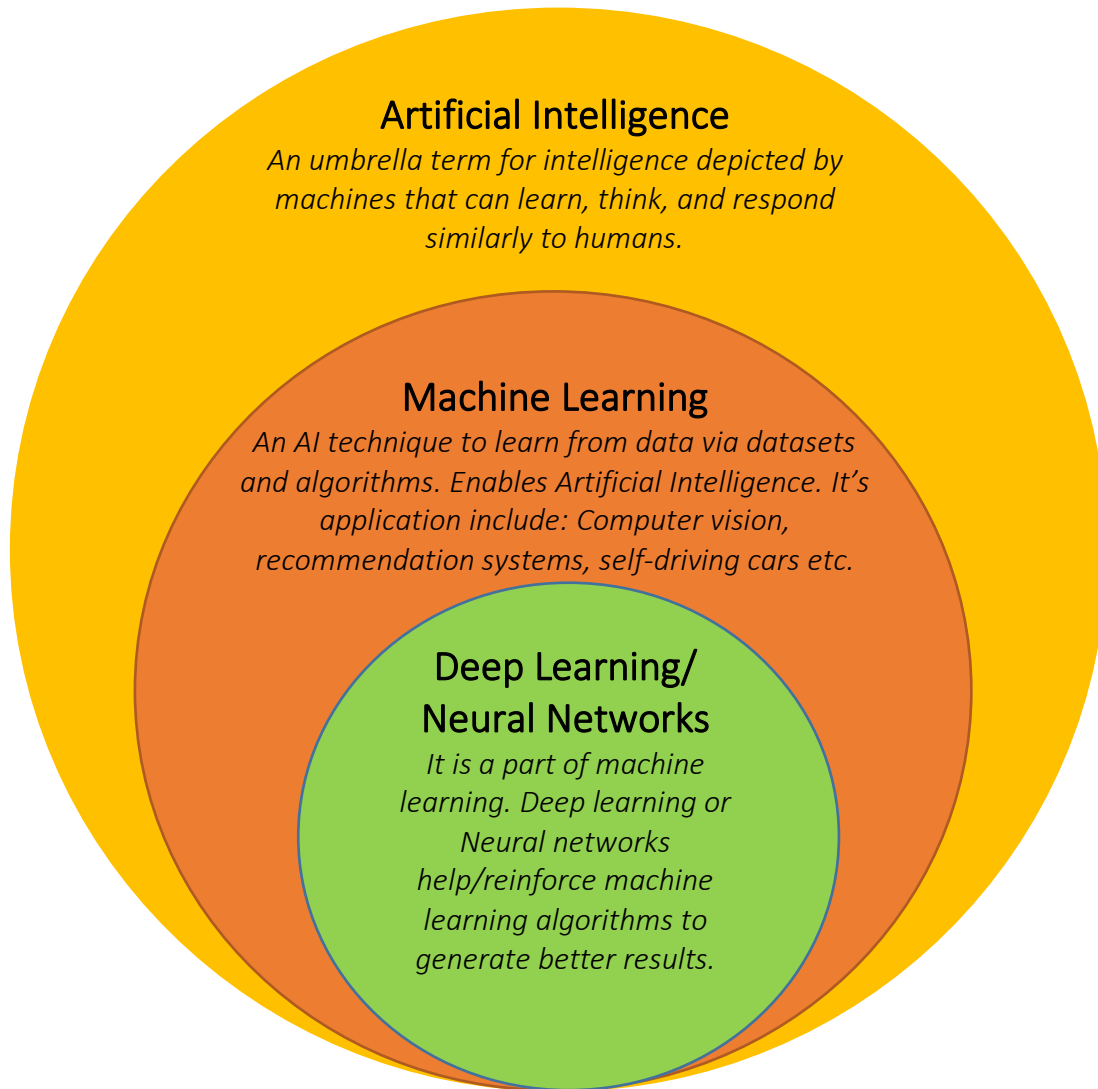


Figure 10: Artificial Intelligence vs. Machine Learning vs. Deep Learning

## 2.5 Machine Learning

We see how AI is all about mimicking human intelligence. However, the question is how is this achieved till a certain point? As implied earlier, machine learning is used to accomplish this. In most cases, when AI is discussed, it very often means the machine learning aspect. The terms are used interchangeably. A popular definition of machine learning is given by Arthur Samuel 1959, a pioneer of AI:

*“Field of Study that allows computers to learn without being explicitly programmed [36].”*

Machine learning is transforming almost every industry. Machine learning works on two things: data and machine learning algorithms.

As we have gathered much data over the years, we can now use this data to build AI systems to drive business value. However, it is essential to point out that having much data does not mean AI systems can be created. The amount of data needed depends entirely on the considered project. Moreover, AI cannot do everything. A project must be tested for feasibility in AI. A general rule of thumb or principle for AI projects is that:

*“If a typical person can do a mental task with less than one second of thought, we can probably automate it using AI either now or in the near future. [37]”*

Looking for a defect in a plate takes, in most cases, less than a second; this can be automated using AI, and we see this application in the manufacturing industry. Furthermore, to automate this task using AI, we use gigabytes of data (labelled data in this context) consisting of pictures of plates labelled as “good” and “defect” and train the AI model on this using machine learning algorithms such as linear regression, logistic regression, SoftMax regression. The choice of algorithm (linear regression, logistic regression, SoftMax regression) or the ML model (supervised learning, unsupervised learning, reinforced learning) depends entirely on what output is desired from the input data.

In a machine learning project, data is collected first, then choosing the most appropriate algorithm and training the ML model on the data to make predictions and/or recognize patterns. Researchers at MIT well put the functionality of machine learning: *“The function of a machine learning system can be **descriptive**, meaning that the system uses the data to explain what happened; **predictive**, meaning the system uses the data to predict what will happen; or **prescriptive**, meaning the system will use the data to make suggestions about what action to take [38].”*

## 2.6 Types of Machine Learning:

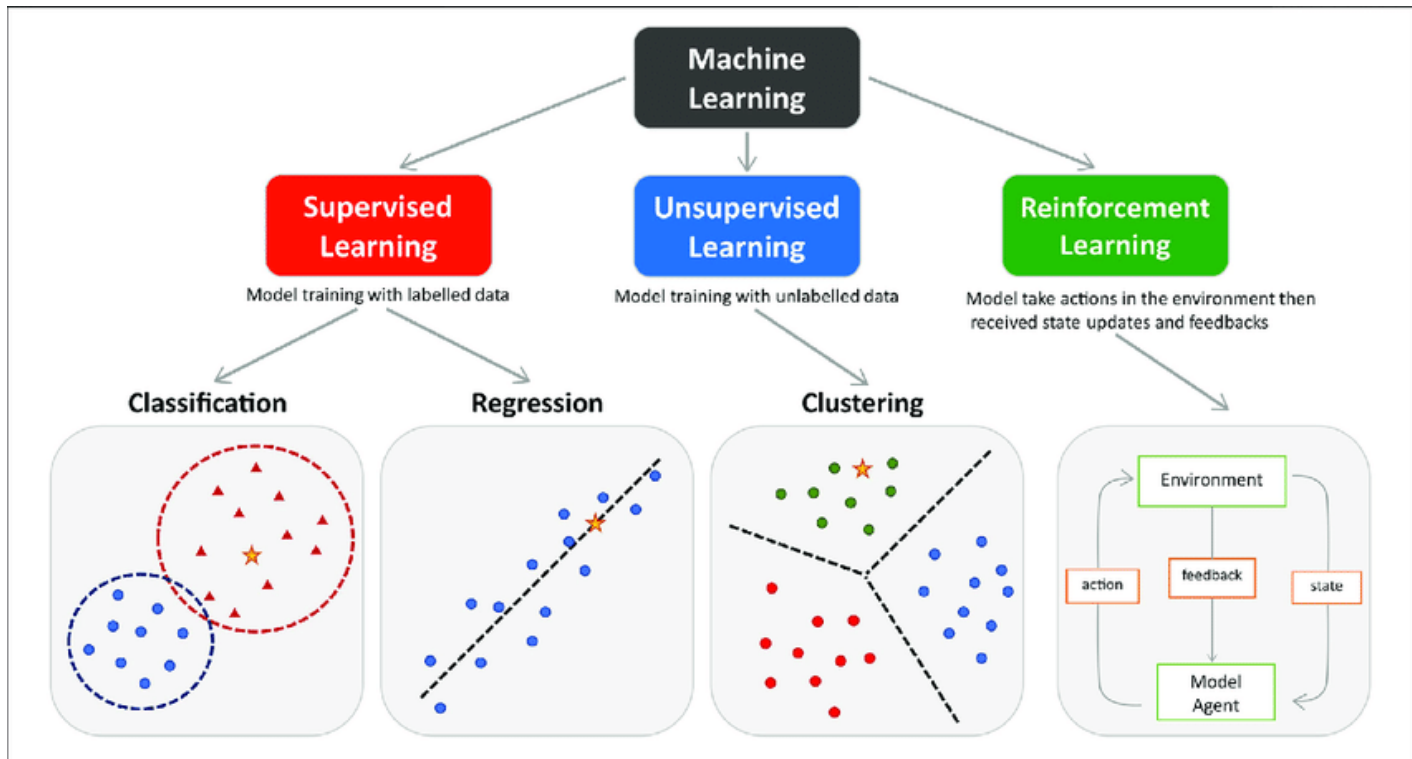


Figure 11: Types of machine learning [39]. Figure available via license: [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/)

### 2.6.1 Supervised Machine Learning

In supervised machine learning, the model learns from a labelled dataset. A dataset is constructed where each element of the dataset is labelled manually in most cases. The model learns the input variable  $X$  and the corresponding output variable  $Y$  and then makes predictions based on what it learned. It learns from input-to-output mappings [40]. The model is evaluated against a subset of its dataset that has yet to be used in training to see how accurate its predicted results are. A model needs to generalize (*“ability to adapt properly to new, previously unseen data, drawn from the same distribution as the one used to create the model.”* [41]) well on the test dataset to be approved as a good machine learning model.

For example, a labelled fruits dataset is constructed to develop a supervised machine-learning model to recognize fruits, as shown in figure 12. The dataset can be split to train data and test data. The model is then trained on this labelled data to identify fruits in the dataset. It is then evaluated using test data to see how accurately it identifies the fruits in input test data.

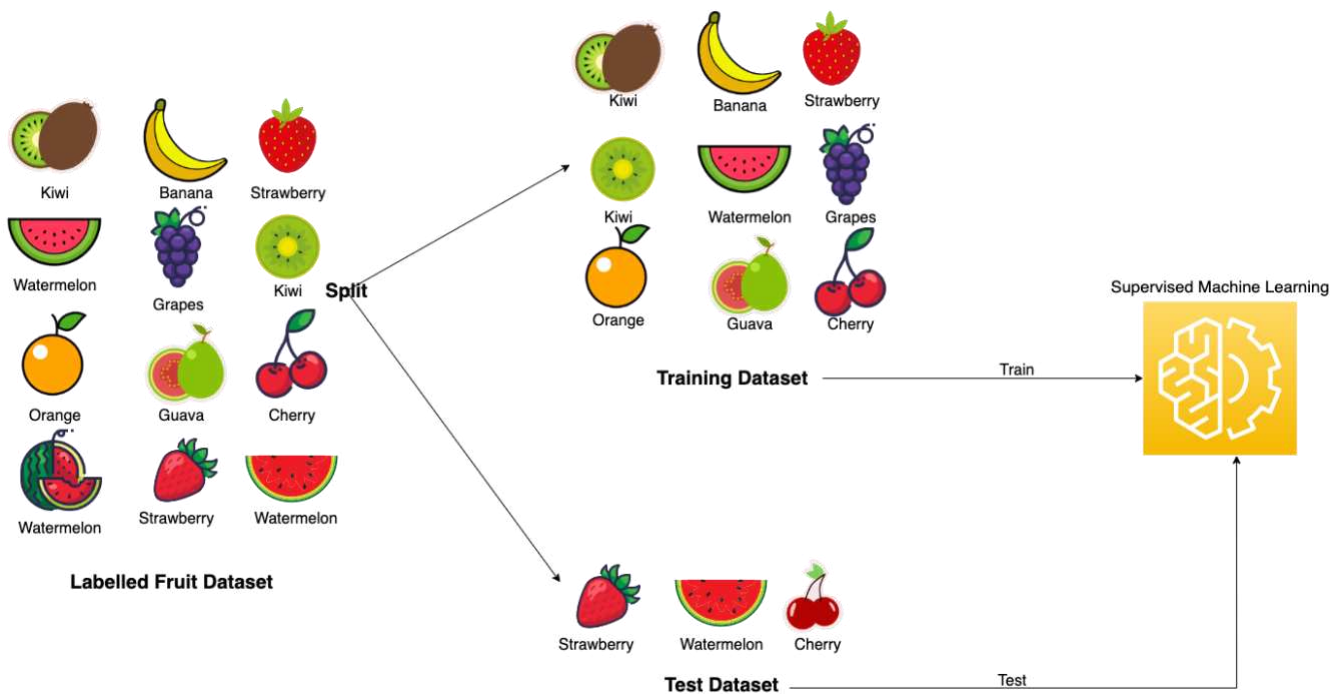


Figure 12: Supervised Machine Learning

The number of images used in a typical dataset is multiplied, and the photos are realistic depending on their application. The cartoon images are just used for illustration purposes.

Elaborating on the steps in supervised machine learning:

## 1. Collecting Data

This is an essential step in machine learning. Data must be collected from reliable sources. Data must be regularly updated to keep our ML model updated; not many empty column values must be in the dataset, as this results in performance issues during training. The quality of data directly affects the performance and predictions of the ML model. If the ML model must make relevant and correct predictions, it must be trained on a corresponding dataset [42].

## 2. Preparing the Data

The data is prepared by cleaning it – removing unwanted rows and columns (also known as feature selection), filling in the missing values, and converting datatypes. Data cleansing needs to be performed as implied; it affects the performance of the algorithms. Data can also be visualized in the process to understand the relations present between the variables and classes. The data is then split into training and testing data (for example, 75% can be training data and 25% can be the test data -75:25 is often used). The model is trained using the training dataset, and the model's

performance is evaluated against the test dataset, which helps us check the model's accuracy [42].

Data normalization is also a data preparation technique – it changes the values of numeric columns to a standard scale, usually between 0 and 1. Data normalization is done to reduce the differences between the numeric values. [43, 44]

### 3. Choosing a Model

As mentioned earlier, the model is chosen depending on the desired output. Suppose a patient must be **classified** as having a brain tumour; a classification algorithm like logistic regression can be used. If the sale of a product in a specific place must be predicted, linear regression can be used. The algorithms of linear regression and logistic regression will be discussed shortly.

So a model can be chosen according to what the machine learning system must produce as an output as it addresses the problem. The selection is also based on available resources [44].

### 4. Evaluating a Model

Then the model is trained on the training dataset to generate results corresponding to our model choice. The model is then evaluated by being tested on unseen data. The model's performance on this unseen data helps properly evaluate the model. Testing an ML model on previously seen data will undoubtedly produce high accuracy, but this differs from the evaluation standard. An ML model is evaluated to see how it works with unseen data [44] [42].

### 5. Deploying the Model

The model can then be deployed into production or used in real-time [44].

Types of Supervised Learning:

- Regression

Regression algorithms determine the relationship between the independent variable, i.e., the input variable X and the output variable Y or the dependent variable [45]. The input variable X may consist of many features, and these features may directly or indirectly determine the output Y. Hence Y is the dependent variable. For example, the price of a car is determined by the seating capacity, features (heating, cooling, electric, hybrid), and mileage. The price is the dependent variable Y, and all the features, their absence or presence determining the cost constitute the independent variable or input variable X. A machine learning model predicts the output variable Y given the input features X.

Seating Capacity
Mileage
Electric
Hybrid
Gas
....
n

} Input variable  
X

Linear regression, regression trees, and non-linear regression are some regression algorithms [40].

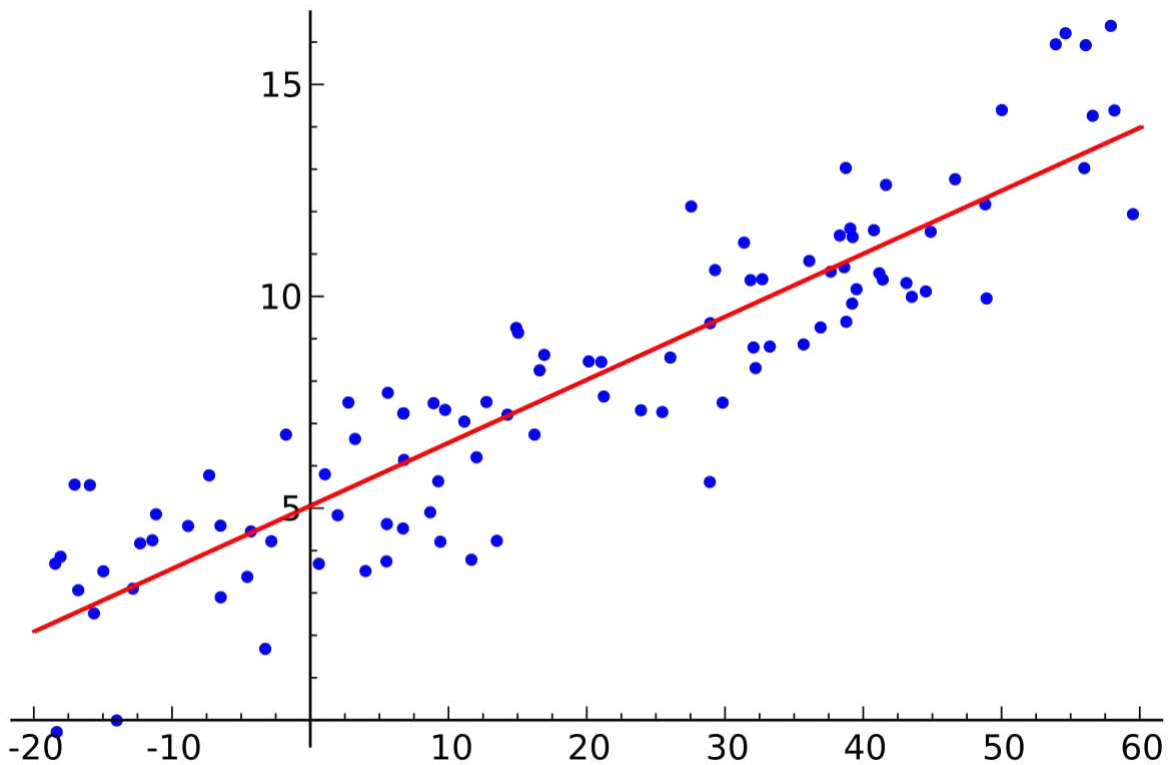


Figure 13: Linear Regression [46]

This algorithm depicts the cause-and-effect relationship between variables X and Y [46]. In machine learning, the algorithm tries to find the best-fit line (the red line in figure 13) [46] to the given dataset, covering sufficient data points to generalize well. The X-axis constitutes the features, and the Y-axis is the dependent variable whose values are determined by the input variable X. The blue dots in figure 13 are data points plotted for each training example or row in the dataset.

- Classification

In machine learning, classification algorithms are used in problems where the input variables must be categorized into a particular class, the output variable. Y variable predicts the class of the input variable X. For example, a machine learning model can be trained to classify images of animals into two categories -cat and dog.

Logistic regression, random forests, and decision trees are examples of classification algorithms [40].

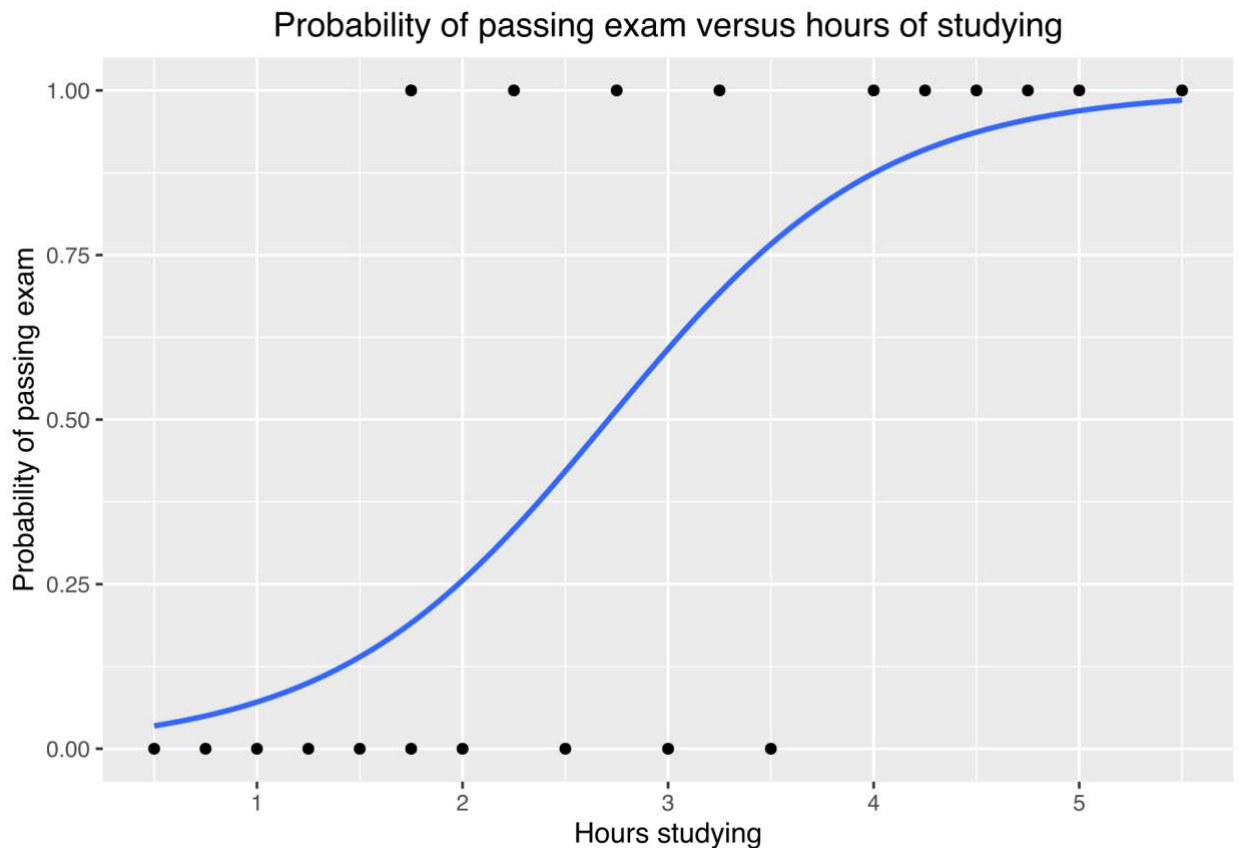


Figure 14: Logistic Regression [47]

In Logistic Regression, the probability of an input belonging to a particular class is calculated [48]. Hence, as indicated in figure 14, the probabilistic output Y (the likelihood of passing the exam) lies between 0 and 1. The above example calculates the chances of passing an exam depending on the study hours.

## 2.6.2 Unsupervised Machine Learning

Patterns are recognized within the data using machine learning. It is seeing what machine learning can do with unlabelled data. The algorithms are designed to find hidden and interesting patterns within the dataset. Their results can then be used for exploratory data analysis, cross-selling strategies, customer segmentation, and image recognition [49].

- **Clustering**

Clustering is an unsupervised machine-learning algorithm that groups data together based on their similarities and differences [49].

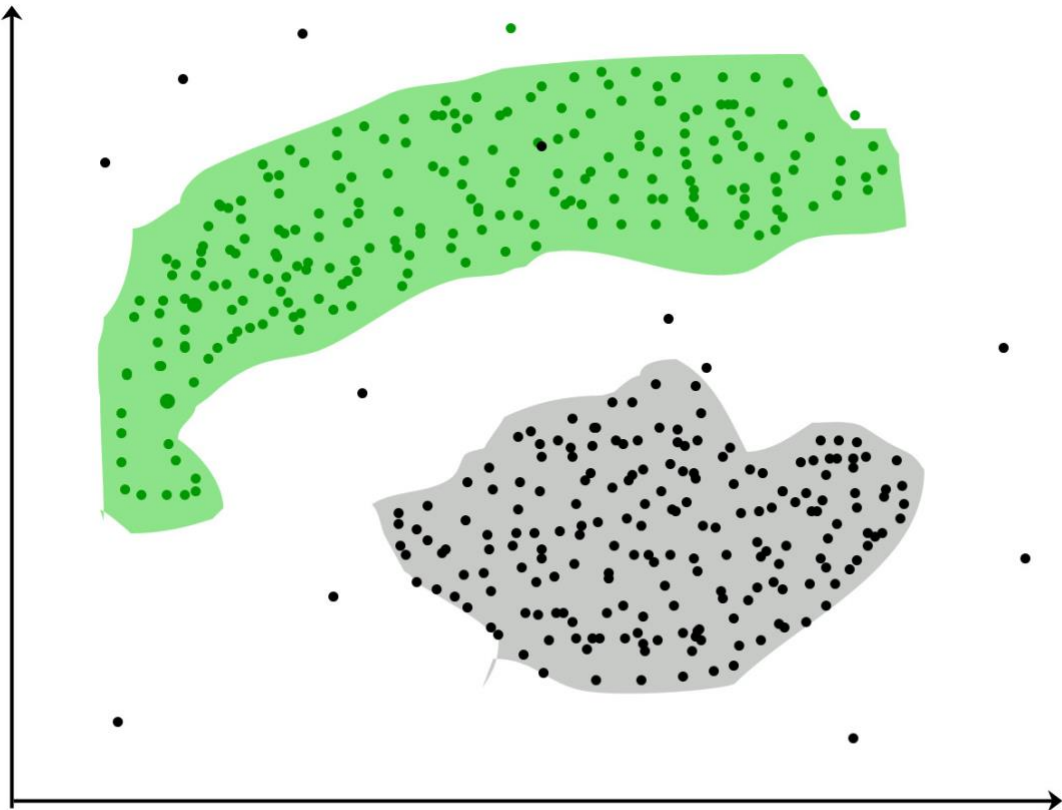


Figure 15: Unsupervised Machine Learning: Clustering [50]



### 2.6.3 Reinforced Learning

It is an algorithm in which the model learns in trial-and-error methods. It is rewarded for acceptable behaviour and certain indicators may be used to discourage unacceptable behaviour. This way learning is reinforced by rewarding good behaviour. As indicated in figure 16, a learning agent is trained by placing it in a state by feeding it all the information required to produce an output or act in a given environment. Depending on the action it takes, it is either rewarded or a penalty is given. The reward could be a positive number, and a negative number can be assigned to discourage unwanted or irrelevant action. The agent is programmed to increase its rewards [51].

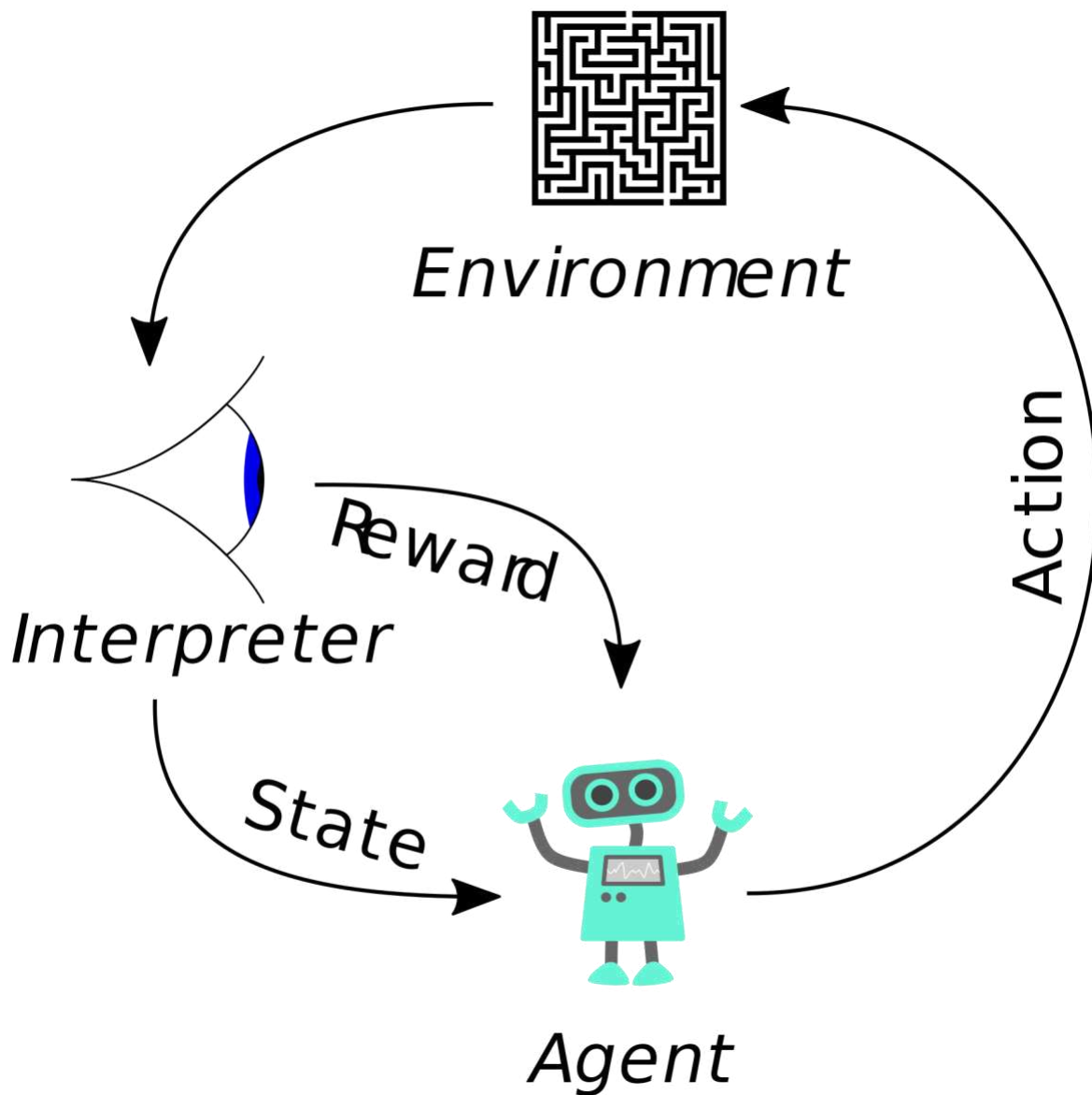


Figure 16: Reinforced Learning [52]

## 2.7 Deep Learning/ Neural Networks in Machine learning

### 2.7.1 Neural Network

- Neural Networks are used in machine learning to perform complex calculations, like recognizing faces and summarizing documents, with greater accuracy [53].
- The neural network consists of artificial neurons, as shown in figure 18. These neurons form the units of the neural network. A neural net consists of thousands of these artificial neurons or “processing nodes” that are interconnected to each other [54].
- The human nervous system inspires neural networks. As in the human body, the messages are passed from one neuron to another via electrical signals. In artificial intelligence, these electrical signals can be attributed to activations within a neural network. The output produced by the activation functions of the neural network layers is passed as an input to the succeeding layers of the neural network.

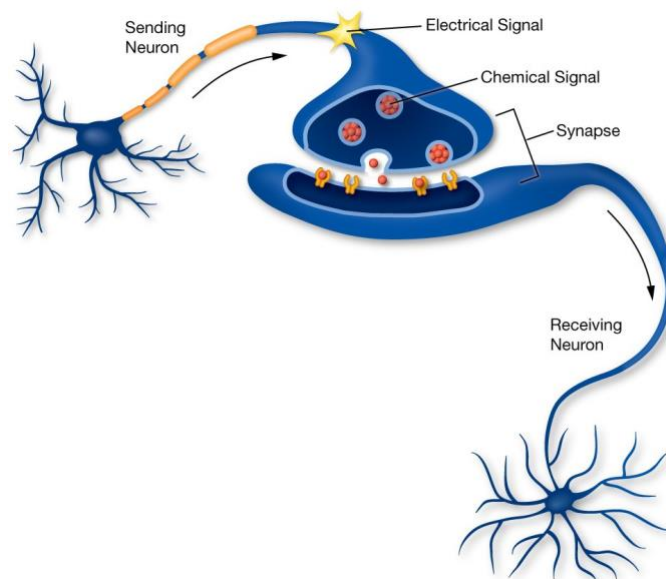


Figure 17: A Biological Neuron [55]

- Most Neural Networks feed-forward or use forward propagation, where the data flow from input to output layers. They can be programmed to do backpropagation. [56]
- In artificial intelligence, weights and biases, which are usually random values, are used to make predictions. The values of weights and biases can be manipulated or set to a particular value to produce the desired output/increase the model's accuracy. In the backpropagation method, the error (the difference between the resulting output and the desired output) in the output produced by the neural network is calculated, and how the error is associated with each neuron is known, according to which the weights of the corresponding neurons or neuron can be tweaked to produce a better output.

The above process is repeated until the neural network achieves the desired output [57, 56].

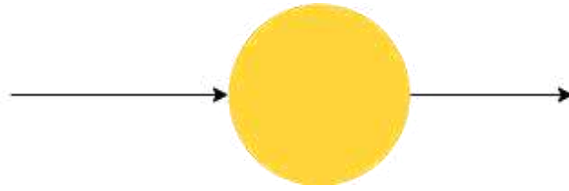


Figure 18: Artificial Neuron

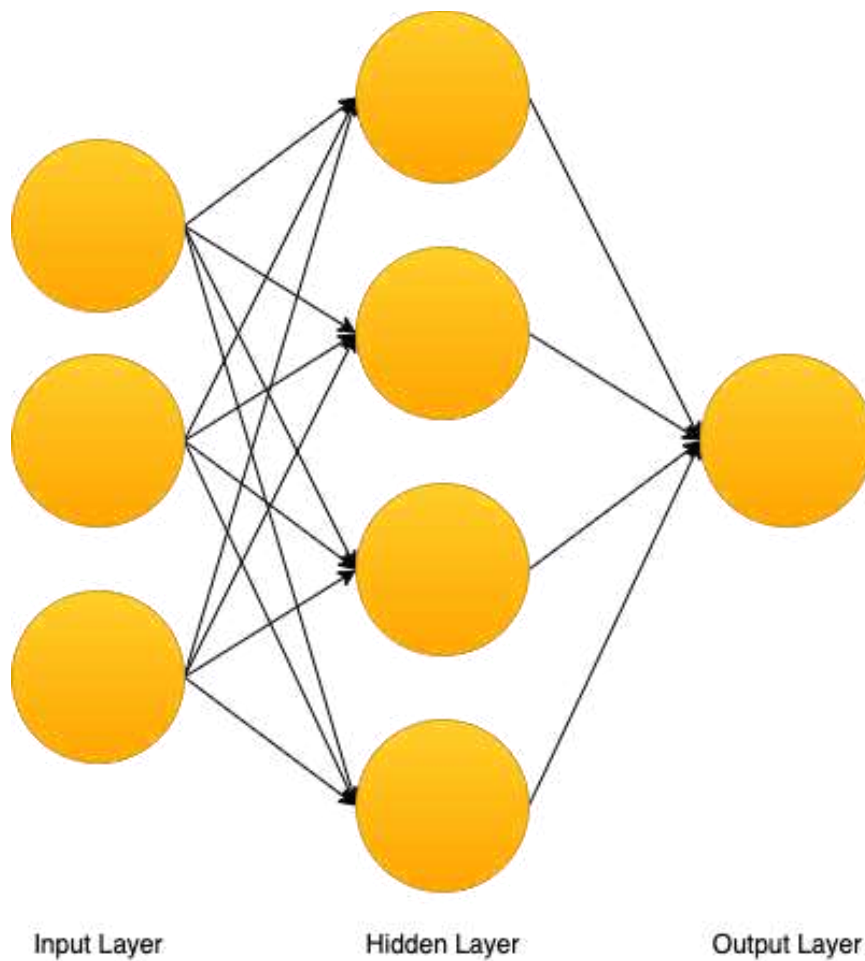


Figure 19: A Neural Network

## 2.7.2 Convolutional Neural Network

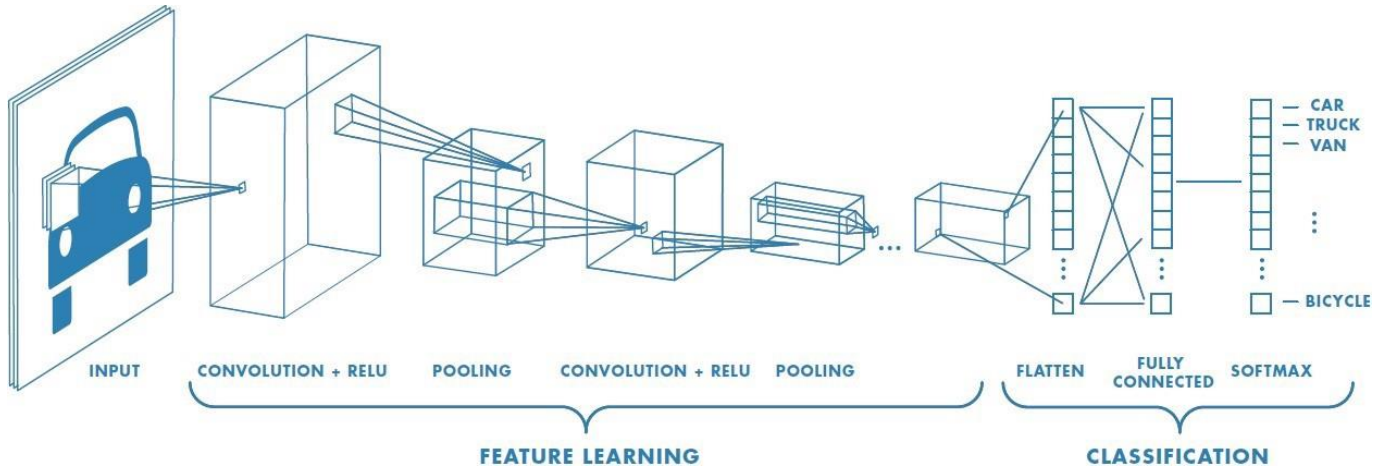


Figure 20: Convolutional Neural Network [58]

- A convolutional neural network, ConvNet or CNN, comprises convolutional layers succeeded by a neural network, as shown in figure 20. The convolutional layers take the input image parts covering different regions looking for features. In the pooling layer, the output produced by the convolutional layer is downsized. After going through the convolutional and pooling layers, the output produced is fed to the fully connected layer or the neural net, where each node is connected to the other to classify the image [59].

## 2.7.3 Deep Learning

- Deep learning is a constitute of machine learning.
- A neural network of several hidden layers achieves deep learning. A neural network qualifies to be a deep learning algorithm or a deep neural network (DNN) if it has more than three layers, including input and output. Figure 21 illustrates the deep neural network.
- The input layer is where the input data is fed into the neural net, and the output layer in figure 21 gives an output of eight classes. The output layer depends on the algorithm used in the last layer preceding the output layer. The choice of algorithm directly depends on the problem. If linear regression is used, an output layer consisting of a single neuron is appropriate. The number of outputs equals the number of artificial neurons in the output layer. Eight artificial neurons must be used if we have eight outcomes or classes for a classification problem.

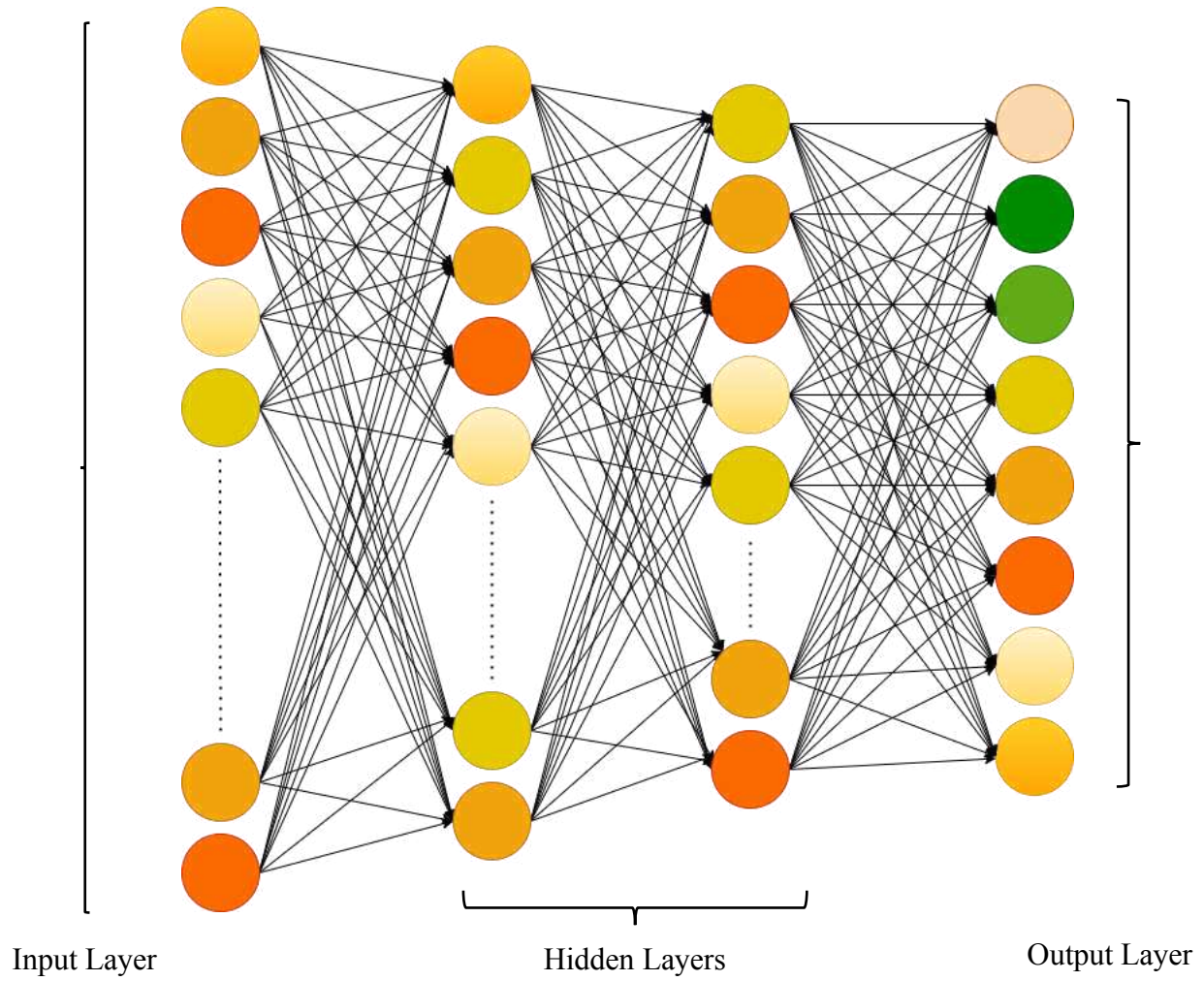


Figure 21: Deep Neural Network

Few concepts worth mentioning as they are implemented in chapter 4 of this report:

## 2.8 Confusion matrix [60]

A confusion matrix helps evaluate the performance of a classification algorithm. For a binary classification problem, the confusion matrix looks as depicted in figure 22. For example, assume 165 samples ( $n = 165$ ) or benign and malware applications. Samples of these 60 are benign applications, and 105 are malicious samples. In figure 23, the predicted malicious application is 55, and benign applications are 110.

- The TN denotes True Negative: where the value is predicted, and the actual value is the same, which is 50 in figure 23. It correctly predicted 50 benign applications to be benign.
- FP denotes False Positives. In figure 23, the classification algorithm predicted ten benign samples as malware.
- FN denotes False Negatives. The classification predicted five malicious samples to be benign.
- TP denotes True Positives; like in true negatives, the actual value equals the predicted value. In figure 23, the classification algorithm rightly predicted the 100 malicious samples.

		<b>Predicted:</b> <b>NO</b>	<b>Predicted:</b> <b>YES</b>
n=165			
<b>Actual:</b> <b>NO</b>		50	10
<b>Actual:</b> <b>YES</b>		5	100

Figure 22: A confusion matrix [60]

		<b>Predicted:</b> <b>NO</b>	<b>Predicted:</b> <b>YES</b>	
n=165				
<b>Actual:</b> <b>NO</b>		TN = 50	FP = 10	60
<b>Actual:</b> <b>YES</b>		FN = 5	TP = 100	105
		55	110	

Figure 23: Elaborate confusion matrix [60]

## 2.9 Cross-validation techniques

Cross-validation techniques ensure the data can make predictions close to actual values (generalization) after training.

Some standard techniques are: [61]

- **Hold-out cross-validation:**



Figure 24: Hold-out cross-validation [61]

The holdout cross-validation technique splits the dataset into train and test sets. The machine learning algorithm is trained on the training dataset and is then tested on the remaining test set, which is the data it has not seen before. Its performance is evaluated to be good or bad based on how well it was able to generalize on the test. The data is usually split into 80% training data and 20% testing data.

- K-fold cross-validation

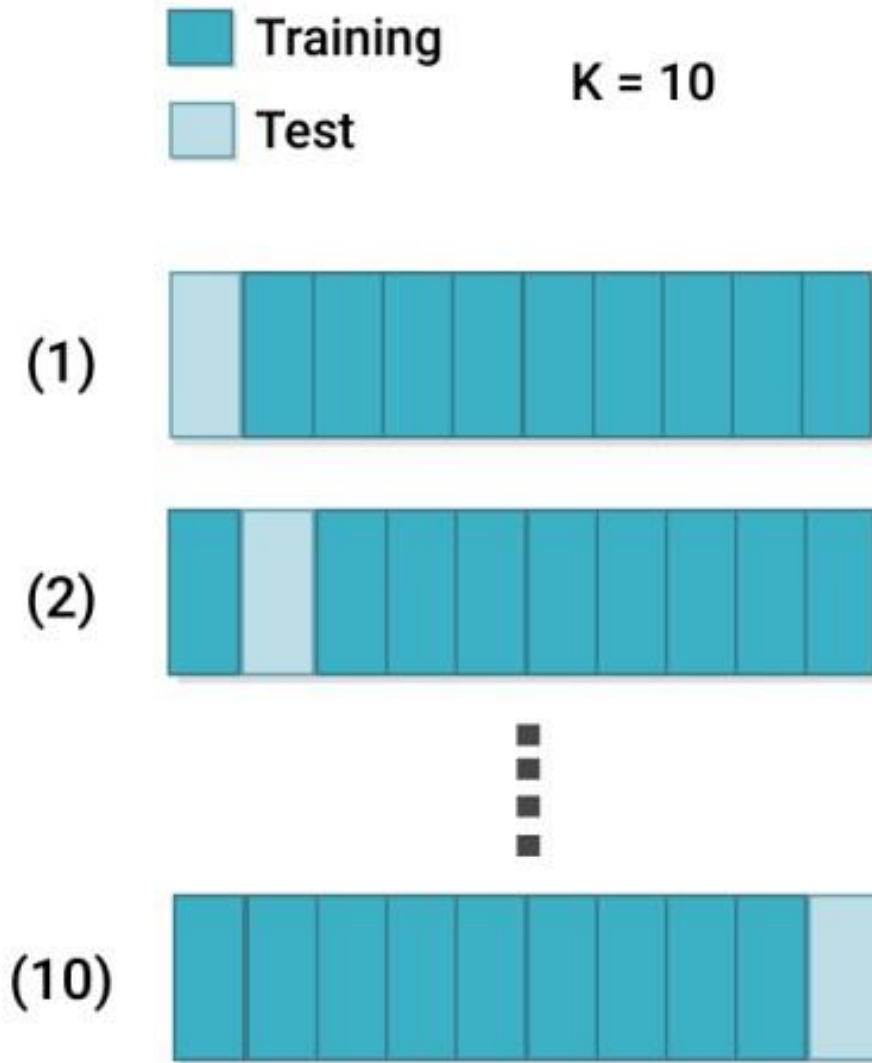


Figure 25: K-fold cross-validation [61]

In K-fold cross-validation, the data is split equally to the number assigned to k. If  $K=10$ , the dataset is split into ten partitions known as folds. After splitting,  $k-1$  folds are used to train the model, and the remaining fold is used to test the model. After each iteration, the fold used for testing changes from the first to the second fold in the second iteration, as depicted in the figure.



## 2.10 Generalization, Overfitting, Underfitting

- Generalization refers to the machine algorithm's ability to make predictions on relevant data it has not seen before, close to the actual values.
- When the machine algorithm performs too or exceptionally well with high accuracy on the training data but cannot make predictions or generalize well, it would mean that the algorithm is **overfitting**.
- When the model is not training well and obviously unable to generalize well, the algorithm is said to have a problem with **underfitting**.

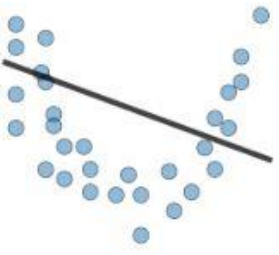
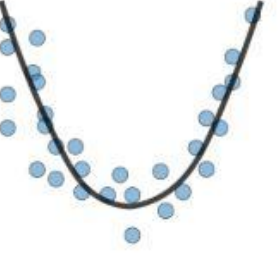

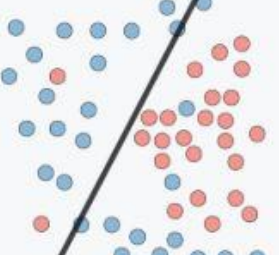
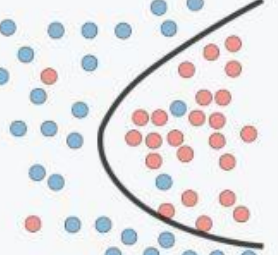
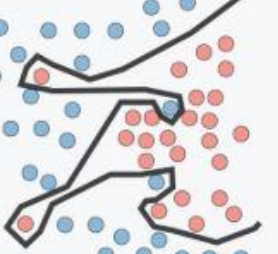
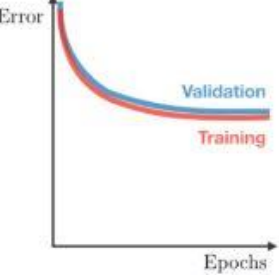
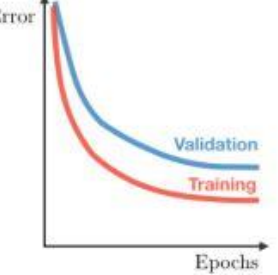
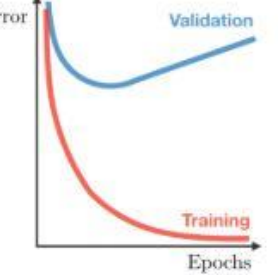
	Underfitting	Just right	Overfitting
Symptoms	<ul style="list-style-type: none"> <li>• High training error</li> <li>• Training error close to test error</li> <li>• High bias</li> </ul>	<ul style="list-style-type: none"> <li>• Training error slightly lower than test error</li> </ul>	<ul style="list-style-type: none"> <li>• Very low training error</li> <li>• Training error much lower than test error</li> <li>• High variance</li> </ul>
Regression illustration			
Classification illustration			
Deep learning illustration			
Possible remedies	<ul style="list-style-type: none"> <li>• Complexify model</li> <li>• Add more features</li> <li>• Train longer</li> </ul>		<ul style="list-style-type: none"> <li>• Perform regularization</li> <li>• Get more data</li> </ul>

Figure 26: Underfitting vs. Just right vs. Overfitting [62]

## 2.11 Advantages and Disadvantages of AI

### 2.11.1 Advantages

#### 1. Reduction in manufacturing error

AI algorithms can be trained to increase their accuracy and precision in making predictions in its various applications [63].

For example, when detecting defects in manufactured products, an AI model performs better than manual detection as it is time-consuming, and it is common to let a defective product or two be bypassed when done manually. AI helps reduction in production errors as it learns from all the previous corresponding production data [64] and is quicker than manual detection. AI can be used to automate this task.

#### 2. AI in medicine

AI is used in diagnosing illnesses. It does not conclude but supports healthcare professionals in coming to a decision. AI is powerful at learning from all the data it is given; an AI model can be trained on medical data – the history of patients, their medical history to recognize patterns and make predictions. It can assist human radiologists in diagnosing accurately by looking at the image data (X-rays, CT-Scans) [65].

#### 3. Enhanced Customer Experience

An AI-powered chatbot can be developed to assist customers the way humans do, giving personalized responses to each customer with the help of natural language processing (NLP). [66]

#### 4. AI in Cybersecurity

AI can be used in cybersecurity to detect malware. Malware dwell times are staggeringly long – an average of 277 days, according to IBM's Cost of a Data Breach 2022 report. AI can help speed up the process of detection. It is also reported that organizations that had fully deployed artificial intelligence and automation programs helped them identify malware 28 days earlier than those that did not and saved USD 3.05 million in costs [22]. AI methodologies are used in heuristic-based malware detection methods to help speed up malware detection and prevent attacks.

## 2.11.2 Disadvantages

### 1. Biased and Discriminatory algorithms

Depending on how datasets are built, AI can be biased. The datasets the machine learning model is trained on or the input data determines the output quality. When training a machine learning model as a machine learning engineer, the hope is that the data we are provided with, or the data found online, is legitimate/genuine without **societal bias**.

A famous example of this undesirable behaviour is COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) – an algorithm used in US court systems to profile an offender as a recidivist (it is a tendency to relapse, especially into criminal behaviour [67]) Because of the nature of the data used, the algorithm falsely classified people belonging to a particular race – the black defendants to be 45% more likely to offend again than the whites whose risk of reoffending was determined to be 23% [68]. Hence it can be concluded that the quality output of the machine-learning model is directly proportional to the data quality.

### 2. Disinformation

This is a case of misuse of AI. AI is being used to generate fake data using genuine entities. In 2020 an activist group called Extinction Rebellion produced a fictional speech by Belgian Prime Minister Sophie Wilmès. They created it using an original video of the prime minister and used AI to modify the words [69].

This is known as deep fake – where deep learning generates fake data. Also called by the Guardian “21st-century Photoshopping” [70]. This can be used to create misleading information.

### 3. Lack of Transparency

In AI, different models are built, and these models are used. When encountering technical problems while using these models, it takes much work to diagnose them. There needs to be more transparency on how the model is built. An AI model can be diagnosed based on the information available. Like in software, we have various testing methods conducted to look for bugs with AI models *“they are not code per se in that we cannot just examine code to see where the bugs are”* [71].

# Chapter 3: Detect, Prevent, Analyze and Respond

### 3.1 Detect

Malware detection comprises mechanisms to identify and protect against harm from viruses, worms, Trojan horses, spyware, and other forms of malicious code. [72]

Detecting malware is challenging as it increasingly gets better at hiding itself, speaking mainly of the new generation malware. Traditional malware is a term attributed to static malware detonated on a general level (not targeted). *Next-generation* malware is “*the malware which can run in kernel mode and is more destructive and harder to detect than traditional malware can be defined as new generation malware (next-generation).*” The New Generation malware targets devices to infect is persistent and changes forms.

The reasons malware detection is challenging:

**Encryption:** Encrypting the code's malicious block and regular code becomes complicated for the computer to detect the malware unless decrypted [73].

**Oligomorphic:** Different keys are used when encrypting and decrypting malware [73].

**Polymorphic:** Like Oligomorphic, different keys for encryption and decryption are used, but it uses decryption techniques to change its form, appearing to be a different file or changing file signatures. It transforms into a different file once after downloaded into the system. Even if the previous malicious file has been detected, the mutated new file still exists on the system [73, 74].

**Metamorphic:** Metamorphic does not use encryption, but like polymorphic malware, it changes forms. The opcode (operation code) of the malicious program is altered after each iteration. In metamorphic malware, the malware rewrites its code without using decryption techniques like in polymorphic. The malware does not return to its original form [73, 75].

**Stealth:** Malware utilizes techniques to protect itself from detection by anti-malware products. This is attributed to stealth. It can do this by hiding itself in legitimate applications or changing the host system to prevent detection [73, 76].

**Packaging:** An obfuscation technique (“the act of making the code of a program hard to discover or be understood by the computer and human without changing how the program works” [77]) that hides the malicious code of the software by compressing it [73, 78].

An algorithm is yet to be developed to detect malware and is proven to be NP-complete [73].

## 3.2 Importance of Malware Detection on an organizational level.

Malware should be detected because the peril of not detecting it can be very costly for organizations. As pointed out throughout the report, the damage of falling prey to malware attacks is severe. Organizations must pay plenty of money to cyber criminals without the guarantee of getting their data back. Investing in security could be cheaper than paying the ransom, with the risk of losing some data permanently and living with the knowledge of the data being in the hands of cybercriminals who could misuse it. Malware attacks do not simply cause financial damage but cause reputational damage to organizations. Organizations may lose existing and potential customers because of malware attacks. Money lost is gained eventually, but trust is not.

When suspicious activity is detected, it can be analyzed. (See section 3.6)

## 3.3 Malware detection techniques

### **Signature-based detection**

Signatures are unique features of a file. They are like the fingerprints of a file [79]. Each file carries a signature that is unique from other files. Patterns are extracted from malware – its signatures are used to detect the malicious presence of the file bearing these signatures. Signature-based detection has a slight error rate; this is the advantage. The disadvantage is that much time, money and human resources are invested in extracting these signatures from files. This method does not detect unknown malware variants present in a system. [79]. Although extracting signatures is time-consuming, detecting using this method is quick. When anti-virus solutions are updated, the signature-based tables are updated, too, as new malware is discovered, signatures are extracted, and the database is updated. Signature-based detection fails to detect mutable malware, like polymorphic and metamorphic malware [79].

### **Behaviour-based detection**

In this type of detection, the software's behaviour is observed to classify it as malware. An executable file is observed to see its intent in action rather than what it claims [79]. The software is programmed to take an executable file and see how it plays out, like ceasing critical system applications that it does not require access to and invoking processes that are not supposed to be invoked in general [80]. Behaviour-based techniques can help detect unknown and mutating malware – polymorphic variants. The disadvantage is the number of false positives, and taking apart an executable and examining what it is doing is a time-consuming process [79]. Anomaly-based detection and specification-based detection are behaviour-based detection techniques.

## Heuristic Methods of Malware Detection

In this type of detection, machine learning techniques are used to detect malware. Heuristic-based malware detection techniques help overcome the disadvantages of signature-based and behaviour-based detection.

It is discernable to use a classification machine learning algorithm to detect malware. In a classification algorithm, for a file to be classified as malware, it needs to look for the features that the algorithm is trained on. When creating a dataset to detect malware, many types of features can be used to create a dataset. Moreover, these features are used to train machine learning algorithms [79]. Features that can be used are listed below:

- **API Calls:**

Programs use Application Programming Interface (API) calls to send requests to the operating systems. The set of invoked API calls can be used as features for creating the dataset. Observing the API calls an application invokes is an assured way of discovering an application's behaviour [79]. A dataset consisting of API calls is used in the lab implementation part of this report.

- **Opcode:**

Operational code or Opcode is a machine language instruction that specifies the operation (arithmetic, data manipulation, logical operations, and program control) to be performed [81].

- **Control Flow Graph (CFG):**

Control flow graphs are flowchart representations of programs [79].

### 3.4 Prevent

- Technically, for malware to be prevented from detonating, it must first be detected. In real-time, when the system raises flags or alerts based on system anomalies, the source application needs to be quarantined and its activity blocked. It must be contained the moment it is detected, and malware is to be extracted to prevent malware attacks from taking place and spreading through the network.
- **Vulnerability mitigation:** As malware exploits existing system vulnerabilities to launch an attack, vulnerabilities within an existing system need to mitigate to prevent malware attacks. Vulnerabilities can be mitigated by patching current software systems and keeping the vendor software up to date [82].
- **Policies:** Policies must be in place to prevent malware attacks. Policies add a layer of security. Examples include:
  1. Employees must be present on-site rather than via email to change banking information. This prevents phishing attacks that request banking information change requests sent via emails.
  2. Scanning of email attachments sent via email [82].
  3. Prohibiting the use of removable media [82].
- **Awareness:** Conducting awareness programs within the organizations instills in employees a general knowledge of malware attacks and why they are essential.
- **Anti-Virus Software:** Anti-Virus software must be installed that scans critical host components, files, and downloads for any known malware and monitors real-time activities [82].
- **Intrusion prevention systems (IPS):** It is a network security tool (that performs packet sniffing [82]) to monitor the networks for any suspected malicious activity. It goes beyond just alerting the security analyst about any activity. They are also known as Next Generation Firewall (NGFW) or Unified Threat Management (UTM) solutions. It is placed in line with network traffic and behind the firewall. Once the IPS detects malicious activity, it can block the source, drop packets, and alert the security analyst [83].
- Running suspected applications within a sandbox before deploying them on the system is a prevention technique.
- Deploying zero-trust architecture where every external application is integrated into the system, user executing processes, and user access rights are to be all “implicitly” validated at every stage [84].



## 3.5 Analyse

Software needs to be analyzed as malicious to be categorized as one.

The suspicious files are analyzed within a network using static, dynamic, or full reverse engineering analysis types. This is important as it aids in the detection and prevention of malware from persisting within the environment [85]. The earlier the malware is detected, the earlier the organization can be alleviated.

- **Static analysis**

In static analysis, an IT professional looks for any abnormalities in the file and manually goes through it without executing it to analyze it as malicious or benign [85].

- **Dynamic analysis**

The suspected application runs in a virtual environment, such as a sandbox, to see how it plays out in that environment. Depending on its behaviour in that environment analyzed by an IT professional, the application can be classified as malicious or benign. Intriguingly, suppose the application displays malicious activity. In that case, its behaviour can be observed to help detect security vulnerabilities in an organization's existing systems and can help mitigate the vulnerability [85].

- **Full reverse engineering:**

Again, a malicious application is taken apart by an IT professional by “*disassembling (and sometimes decompiling)*” its binary code and then converting it to code mnemonics (“*A code that can be remembered comparatively easily and that aids its user in recalling the information it represents [86].*”). Code mnemonics help the professional understand the application's design and learn what it is meant to do. This helps engineer solutions and know the vulnerabilities the malware intended to exploit. Reverse engineering is carried out by various ranges of tools [85].

## 3.6 Importance of Malware Analysis

Malware analysis helps or assists with: [87]

- Threat Intelligence

As described by TechTarget, “Threat intelligence (aka cyber threat intelligence, commonly abbreviated as threat intel or CTI) is information, usually in the form of **Indicators of Compromise (IoCs)**, that the cybersecurity community uses to identify and match threats [87].” It helps researchers to construct attack patterns to detect or predict any attacks and discover vulnerabilities by studying what vulnerabilities the malware is trying to compromise. Malware analysis is used to gather indicators of compromise. The indicators of compromise include sample hashes of MD5, SHA-256 and network artifacts – domain address and IP addresses.

- Incident Response

Upon detection, malware analysis begins, IOCs are extracted, which can help uncover the infected systems and malware is studied for its capabilities to understand the nature and impact of the attack (how far the attack has propagated and how much critical data has been compromised). This analysis helps with incident response and gives a map of how many systems have been affected and how far into the network it spread and could answer the question of “what is common in all the infected devices” to identify any unpatched vulnerabilities.

- Malware analysis can also help understand the communications between the attacker and the malware-infected system. This can be done by recording suspicious network traffic using products such as Network Detection Responses (NDRs) and decrypting the communication (network traffic) to learn what all the attackers did on the system, essentially what data has been compromised and how it was compromised. Knowing how helps take action to prevent similar attacks in the future.

- Threat Hunting

Indicators of Attacks (IOAs) are any abnormal behaviour of systems in the network. Irregular logins in a system that are only supposed to be logged into at a specific time interval can indicate an attack. Malware analysis equips the security engineer/analyst with the knowledge to hunt for threats more efficiently as they study the malware – how it propagates an attack, what vulnerabilities it exploits, and its mode of operation.

- Malware analysis helps in extracting signatures from malware which thereby helps in signature-based detection.

## 3.7 Response

- After detecting malware, an appropriate response has been executed depending on the malware stage.
- When IPS systems or security devices identify malware, the proper response is to deny its permit into the network and store it in a repository to study its functionality. This could help tighten security systems by knowing any new vulnerabilities identified in the systems by studying the malware – learning what vulnerability it was trying to exploit.
- Once malware is detected post-infection, an incident response plan is put into action, typically identifying infected systems and isolating them from the network to prevent the attack from being propagated.

The incident response includes: [88]

- **Incident response: Preparation**

In the preparation phase, the organization prepares for a malware attack by knowing or educating itself on the possible attack vectors. Training staff on specific attacks, such as phishing attacks, helps prevent malicious attacks. Training should be conducted on what kind of threats they are likely to encounter, what actions to take, and how the issue is to be reported.

Some proactive steps to prevent an attack are:

- System vulnerabilities, when identified, should be mitigated as they could serve as a launchpad for an attack.
- Default passwords must not be used.
- Implementing multi-factor authentication
- Careful configurations of devices – as misconfiguration can serve as another attack launchpad.
- Hiring a hacker to test the organization's environment to uncover any vulnerabilities.
- Rehearsing an incident response plan helps test its effectiveness.

- **Incident response: Detection**

A malware infection is detected by observing any anomalous behaviour. For example, in a ransomware attack, when a typically unencrypted file is found encrypted on the system, this could indicate a ransomware attack.

- **Incident response: Analysis**

In the analysis phase after detection, the type of malware is identified and determining entry-point of the malware is also known as root-cause analysis (RCA). Depending on the nature of the malware (ex: a ransomware variant) and the RCA (entry through email, drive-by-download, system vulnerabilities), the following action steps are determined.

- **Incident response: Containment**

If the system is identified and confirmed to have been infected with malware, the system is promptly isolated – shut down and removed from the network. The isolated system can be used to assist in forensic & sample analysis. Containing it in this manner prevents the spread of malware to other systems of the network.

- **Incident response: Eradication**

This step involves purging all malware-infected systems. Depending on the RCA – if the attack was through email, it is advised of the organization to delete all such emails across the organizational network and isolate systems that have opened the email to check for any indicators of attack in the corresponding system. If the attack originated from a website, the website must be blocked.

- **Incident response: Recovery**

Before this process is implemented, the organization must contain and identify the root cause of the infection. The recovery process includes but is not restricted to patching vulnerabilities discovered through the attack and restoration of data from the backup.

- Once recovered from the incident, the vulnerability must be mitigated to prevent this from happening again. Documenting the incident with how, why, and when can help the organization prepare for the future and remember the lessons learnt.

# Chapter 4: Lab Implementation

## 4.1 Introduction

Malware detection with Anti-virus based on signature-based detection does not work well with unknown threats. AI/ML is a better-performing tool for detecting malware than other techniques prevalent in the industry that generally require human intervention and are time-consuming. In cybersecurity, there is always a potential for zero-day attacks. While this attack can be prevented by constant monitoring/observation of the systems, it is common for organizations to fall prey to it. Looking for patterns and system-behavioural changes through present techniques requires human resources and the assistance of various tools such as sandboxes, honeypots, anti-virus software and time, which is critical in the cybersecurity industry. We need better detection tools with APT (Advanced Persistent Threats- evasive malware). The family of Malware that falls under APTs, once infiltrated, stays in the system for as long as they are not detected carrying out malicious activities without the victim's knowledge – this is their stealth nature. AI does not eliminate the need for human intervention but minimizes it and reduces the workload needed to detect malware by scores. In this lab implementation, a sequential neural network model has been developed and trained using the Softmax Regression algorithm to detect malware based on Windows application programming interface -API (*“a code that helps two different software to communicate and exchange data with each other”* [89]) calls. The model results in various implementations are recorded.

## 4.2 Dataset

An open-access dataset from GitHub [90] has been selected for this implementation; to train the Neural Net on it. The dataset comprises eight categories of Malware and their invoked Windows API (Application Programming Interfaces) calls. The Windows API calls are the input, X-variable, and the category they belong to is the Y-Variable. The numerical representations of the dataset available in the git repository were used. This dataset was mainly chosen for the following reasons:

- Most analysts check to see what malware is doing in their system, and the surest way of finding out is through API calls (the processes it invokes in a system).
- In this dataset, the creator describes how he constructed his dataset :

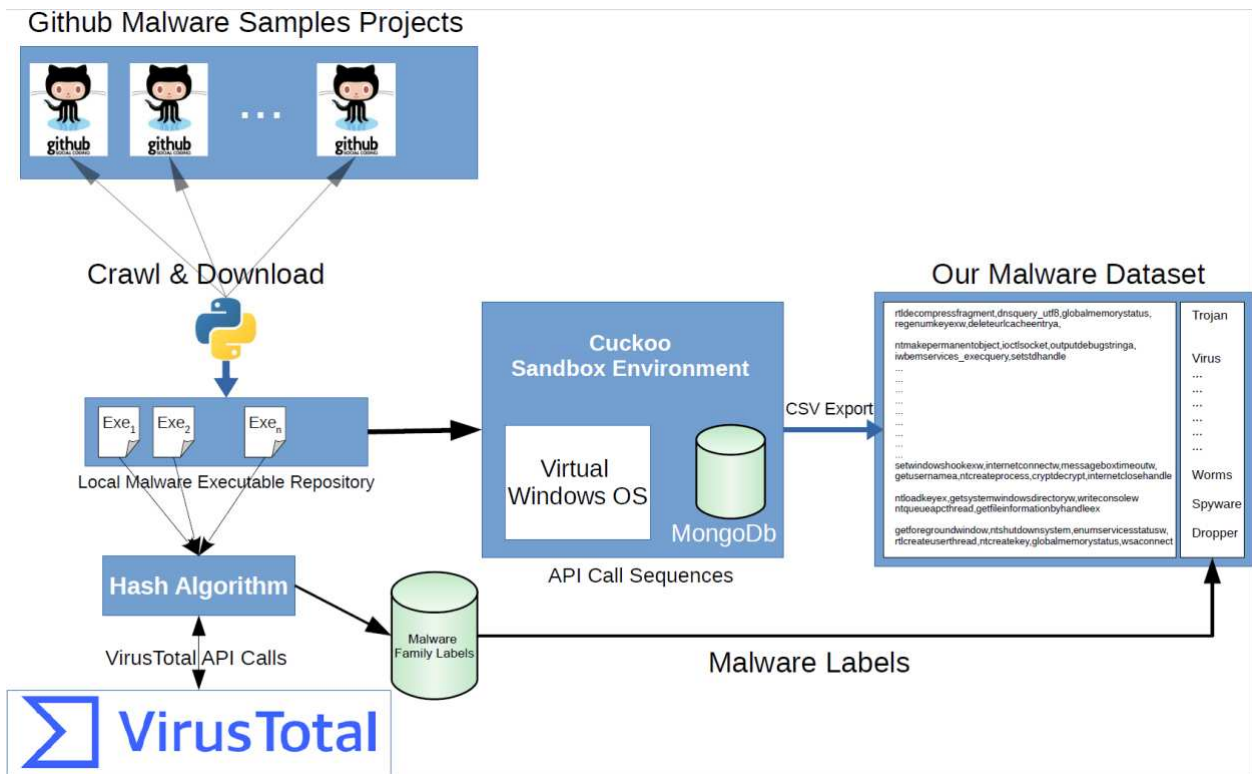


Figure 27: Dataset creation [90]

As shown in the figure, they obtained the MD5 hash values of the malware they collected from GitHub. They searched these hash values using the VirusTotal API and got the families of these malicious software from the reports of 67 different anti-virus software in the VirusTotal [90].

## 4.3 Methods

### 4.3.1 Data

The dataset (Windows API calls dataset) contained a lot of empty columns mainly because each type of malware invoked different sets of windows API calls, and the number of them is different for each malware. The cells containing NaN values were filled with 0. It is necessary to do this as this may result in the neural net not being trained (in this case, the resulting loss while training was NaN). The data values were normalized or scaled to fall within the same range. This helps with the model training. The Y-variable labels were manually changed to numerical values. They were mapped as follows: Virus – 0, Trojan – 1, Backdoor – 2, Downloader – 3, Worms – 4, Spyware – 5, Adware – 6, Dropper – 7, in total eight classes. The labels were then converted to binary to train the multi-class classification model.

### 4.3.2 Classification Model – Sequential DNN (Deep Neural Network) with Softmax Regression

- A Deep Neural Net (DNN) of **sequential** type means: from input to output, passing through a series of neural layers, one after the other [91], was used with the activation function of the output layer as softmax regression to classify the input into one of the eight types of malware. Softmax regression (or multinomial logistic regression) is a generalization of logistic regression to the case where we want to handle multiple classes [92].
- We apply logistic regression algorithms to binary classification problems where the number of categories is only two (true or false, 1 or 0, cat or bird.). We get a probability as an output to see whether the input belongs to a particular class. In a use-case scenario where we are trying to detect if an input image is a horse or not, if the output of our model is 0.8, it means that there is an 80% chance that it is an image of a horse and a 20% percent chance of it not being an image of a horse.
- We apply Softmax Regression in a multi-class classification problem where we have multiple classes (more than two, and the classes are mutually exclusive – i.e., an input can belong to only one class at a time). A probability is assigned to each type of multi-class problem, and naturally, the chances of a Softmax function add up to 1 [93].
- The Softmax Regression formula is given as follows: [93]

$$p(y = j|x) = \frac{e^{(w_j^T x + b_j)}}{\sum_{k \in K} e^{(w_k^T x + b_k)}}$$

$y \in \{1, 2, \dots, K\}$ . In our problem  $K = 8$ .  $w$  and  $b$  are weights and biases.



## Weights and biases:

- Weights and biases are random values that can be tweaked to attain optimum training of the Neural Net.
- *“Weights and bias can be interpreted as a **system of knobs** that we can manipulate to optimize our model — like when we try to tune our radio by turning the knobs to find the desired frequency. **The main difference is that in a neural network, we have hundreds if not thousands of knobs to turn to achieve the result**” [94].*

The Sequential DNN of Implementation 1 is structured as follows:

Model: "model"

Layer (type)	Output Shape	Param #
L1 (Dense)	(None, 200)	3277000
L2 (Dense)	(None, 150)	30150
L3 (Dense)	(None, 100)	15100
L4 (Dense)	(None, 8)	808

=====  
Total params: 3,323,058  
Trainable params: 3,323,058  
Non-trainable params: 0  
=====

The model consists of 4 layers of the type Dense (each neuron is connected to every other neuron in its succeeding or preceding layers). Layer 1 (input layer), Layer 2, Layer 3, and Layer 4 (output layer) have 200, 150, 100, and 8 neurons, respectively. The params are calculated using the formula  $wx+b$  (it is a straight line equation), where  $x$  is the size of the input, and in this lab,  $w$  and  $b$  are the numbers of neurons in each layer; the difference is  $w$  is being multiplied, and  $b$  is added, they affect the magnitude and the position in the dimensional plane correspondingly [94]. The Layer 1 params are calculated as follows:

The input size is 16384, the size of the layer is 200 neurons, and the output is 200 as well so  
 $16384 * 200 + 200 = 3277000$

*“The activation function defines the output of a neuron/node given an input or set of inputs (output of multiple neurons) [95]. It mimics the stimulation of a biological neuron” [95].*

Each layer consists of activation functions. Layer 1 (Input Layer), Layers 2, and 3 have ReLU (Rectified Linear) activation functions, and the output layer has an activation function, Softmax.

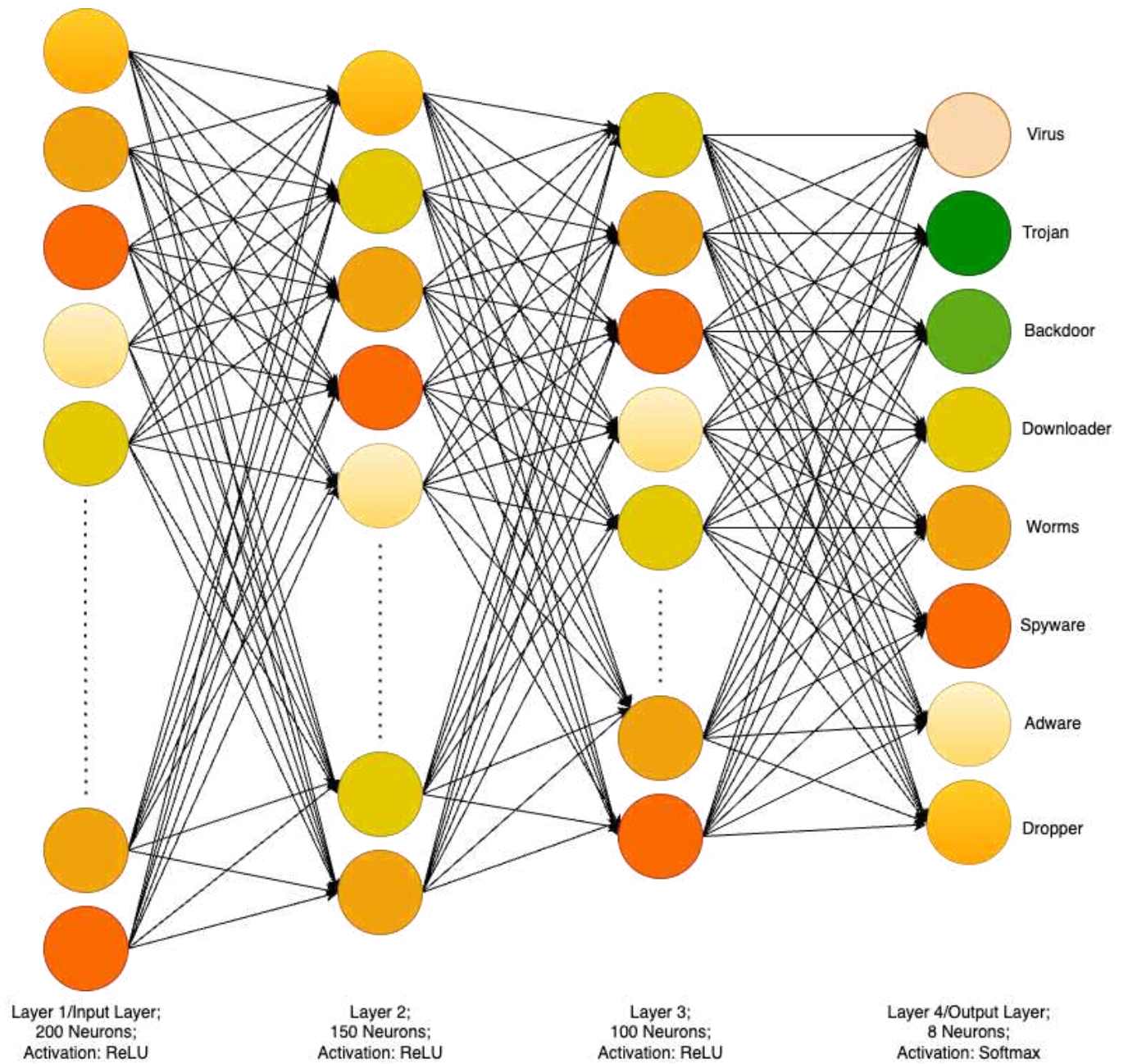


Figure 28: An Illustration of DNN with Softmax Output Layer

### 4.3.3 Explanation

The Neural Net is trained as follows:

1. Model is defined.

```
model = Sequential(  
    [  
        tf.keras.Input (shape=(16384,)),  
        Dense (200, activation = 'relu', name='L1'),  
        Dense (150, activation = 'relu', name='L2'),  
        Dense (100, activation = 'relu', name='L3'),  
        Dense (8, activation = 'softmax', name='L4')  
    ], name = "model"  
) [96]
```

The model uses forward propagation, i.e., input is fed to the neural network and propagated forward as the input produced from each neural network layer is the input to its succeeding layers. The result depends on the activation function being used.

2. The model is compiled using the code:

```
model.compile(loss= 'categorical_crossentropy', optimizer =  
'Adam', metrics=['accuracy']) [96]
```

The **loss function** of the type `categorical_crossentropy` is calculated. The loss function gives us the difference between actual and predicted values. For different algorithms of machine learning, we have other loss functions. Categorical cross-entropy is used for this multi-class classification problem.

**Optimizer** Adam reduces the loss between the actual and the predicted values. It is an algorithm of stochastic gradient descent. In gradient descent, the weights and biases of the neural network are tweaked mathematically by applying differentiation to optimize training and improve prediction results or give the best results by reducing the loss. In stochastic gradient descent, these calculations are done a random manner. Adam is the recommended optimizer.

The **metric** accuracy is used for classification problems to see the fraction of predictions the model got right [97].

3. Then the model is trained using the code:

```
model.fit(X_train, Y_train , validation_data=(X_test, Y_test),  
batch_size=100, epochs=500) [96]
```

The **batch size** indicates the number of training examples it is to run before updating the parameters (weights and biases), and **epochs** indicate the number of times the training data is passed through the model. The batch size is 100, and the training data is passed through the neural net 500 times.

## 4.4. Results and Discussion

### Implementation 1:

- The windows API call (the x variable) dataset and the labels (the y variable) of the corresponding API calls were presumably divided into four CSV files for training and testing purposes. Each CSV file was loaded to variables X\_train, Y\_train, X\_test, and Y\_test correspondingly. No cross-validation method was used, and the results are as follows:

The accuracy plot is shown below in figure 29. A training accuracy of 92% has been attained, and a validation accuracy of 92% has been achieved with this implementation with loss minimizing, as shown in figure 30. A high validation accuracy implies that the model is able to generalize well.

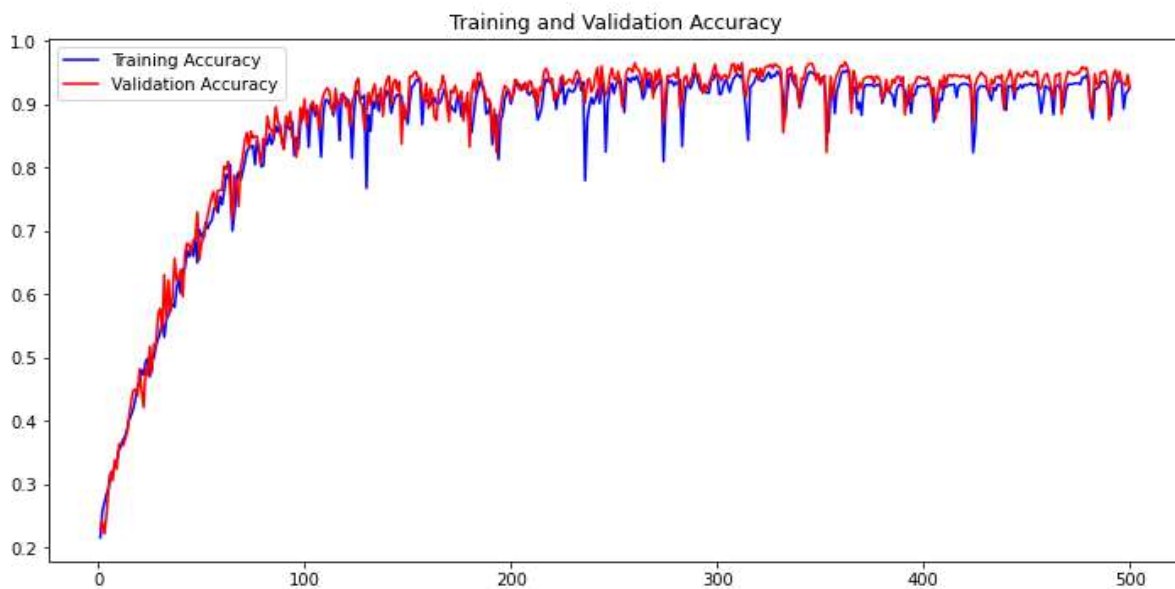


Figure 29: Training and validation accuracy of the neural net

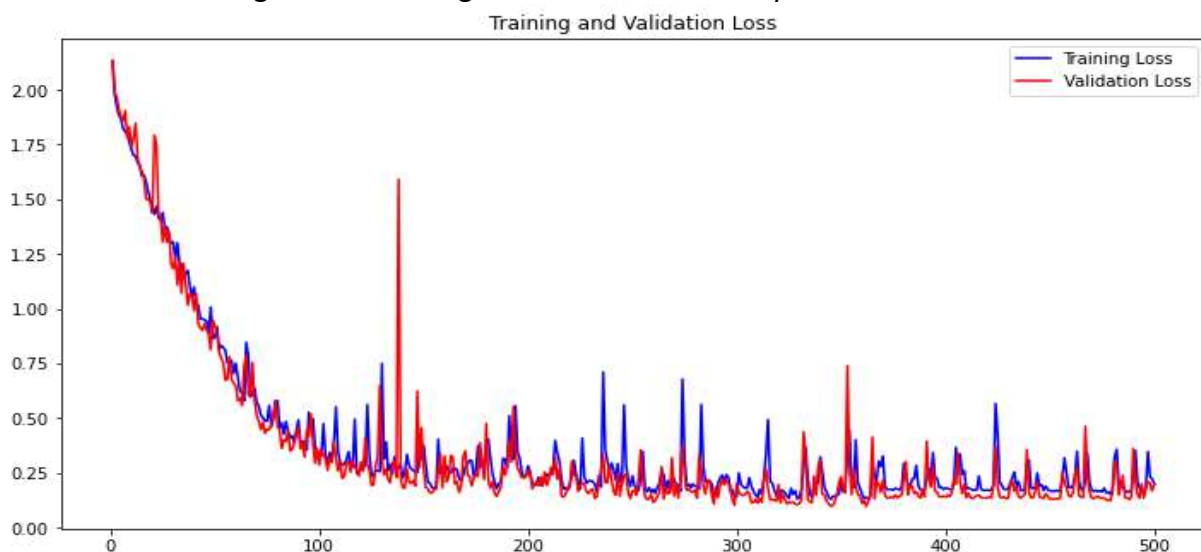


Figure 30: Training and validation loss of the neural net

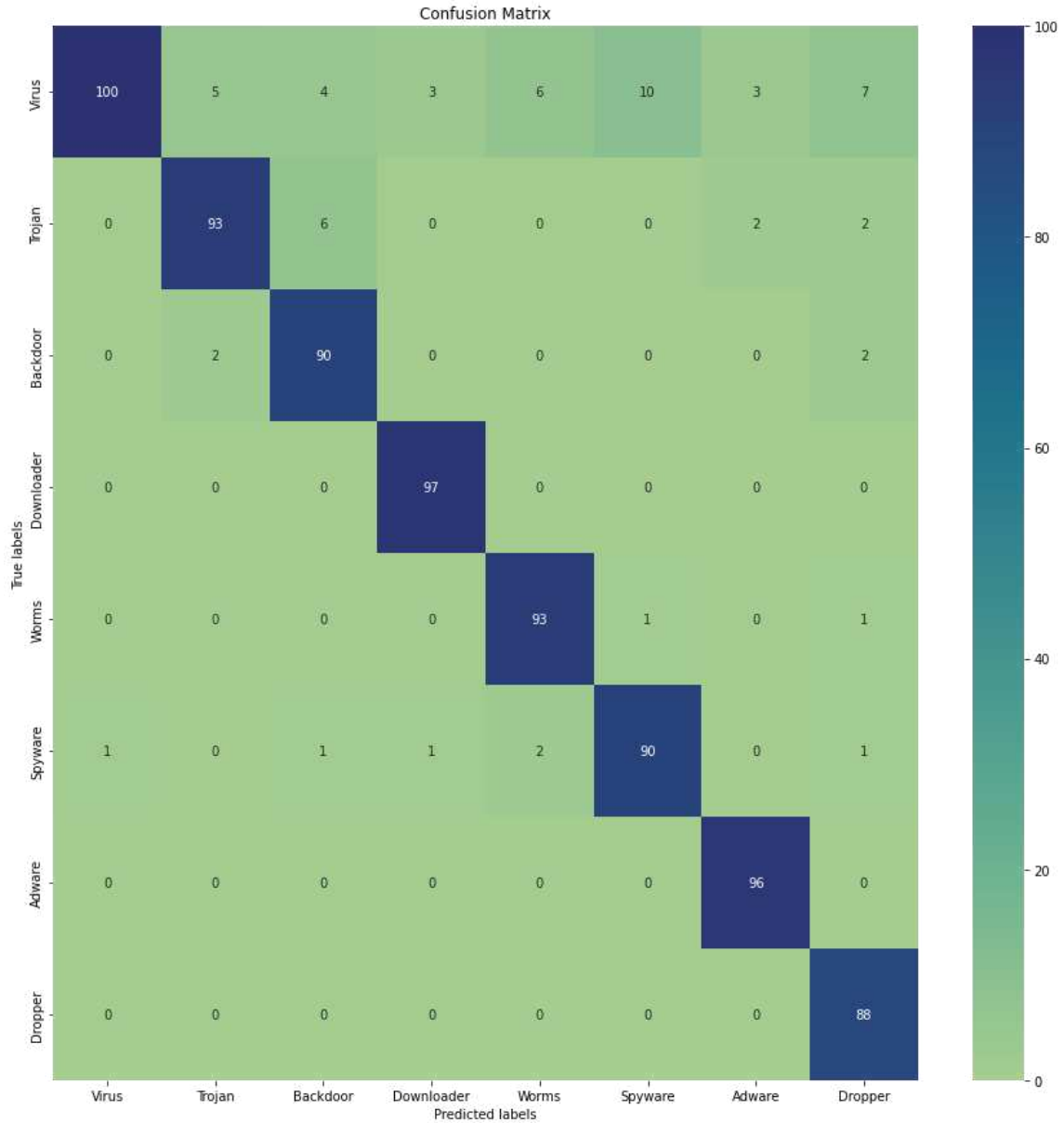


Figure 31: Confusion matrix of Implementation 1

- The test data consists of a total of  $n = 807$  samples. All the values of the confusion matrix add up to 807.
- The sum of the row values corresponding to the class gives the total number of actual samples belonging to that class.
- The total of a class's corresponding column gives that class's total predicted values.

The True Positive (TP), False negative (FN), False Positive (FP) and True Negative (TN) values are calculated as follows: [98]

- The TP value for each class is indicated in dark blue; it is the value at the intersection of the class on the x-axis and the corresponding class on the y-axis.
- The FN value is the sum of the rows corresponding to the class except for the TP value.
- The FP value is the sum of columns corresponding to the class except for the TP value.
- The TN value is the sum of all the rows and columns except the corresponding class row and column values.

1. For Virus,

Total number of Viruses = 138

Predicted Virus = 101

TP = 100

FN = (5+4+3+6+10+3+7) = 38

FP = 0+0+0+0+1+0+0 = 1 (Incorrectly categorized spyware to be a virus)

TN = 668 are not a virus

So, out of 138 actual values, the model predicted 100 virus samples correctly. It predicted the remaining 38 to be Trojan (5), Backdoor (4), Downloader (3), Worms (6), Spyware (10), Adware (3), and Dropper (7).

2. For Trojan,

Total number of Trojans = 103

Predicted Trojan = 100

TP = 93

FN = 10

FP = 7

TN = 697

3. For Backdoor,

Total number of Backdoor = 94

Predicted Backdoor = 101

TP = 90

FN = 4

FP = 11

TN = 702

4. For Downloader,

Total number of Downloader = 97

Predicted Downloader = 101

TP = 97

FN = 0

FP = 4

TN = 706

5. For Worms,  
Total number of Worms = 95  
Predicted Worms = 101  
TP = 93  
FN = 2  
FP = 8  
TN = 704
  
6. For Spyware,  
Total number of Spyware = 96  
Predicted Spyware = 101  
TP = 90  
FN = 6  
FP = 11  
TN = 700
  
7. For Adware,  
Total number of Adware = 96  
Predicted Adware = 101  
TP = 96  
FN = 0  
FP = 5  
TN = 706
  
8. For Dropper,  
Total number of Dropper = 88  
Predicted Dropper = 101  
TP = 88  
FN = 0  
FP = 13  
TN = 706

## Implementation 2:

In implementation 2, only the large 2 CSV files (consisting of the numerical representations – 1000\_calls.zip and 1000\_types.zip) from the git repository [90] were used to train and test the data using the hold-out cross-validation method. The results are as follows:

For hold-out cross-validation:

The model trained with 93% accuracy while it performed poorly or did not generalize well on the test data, i.e., the data it has not seen before.

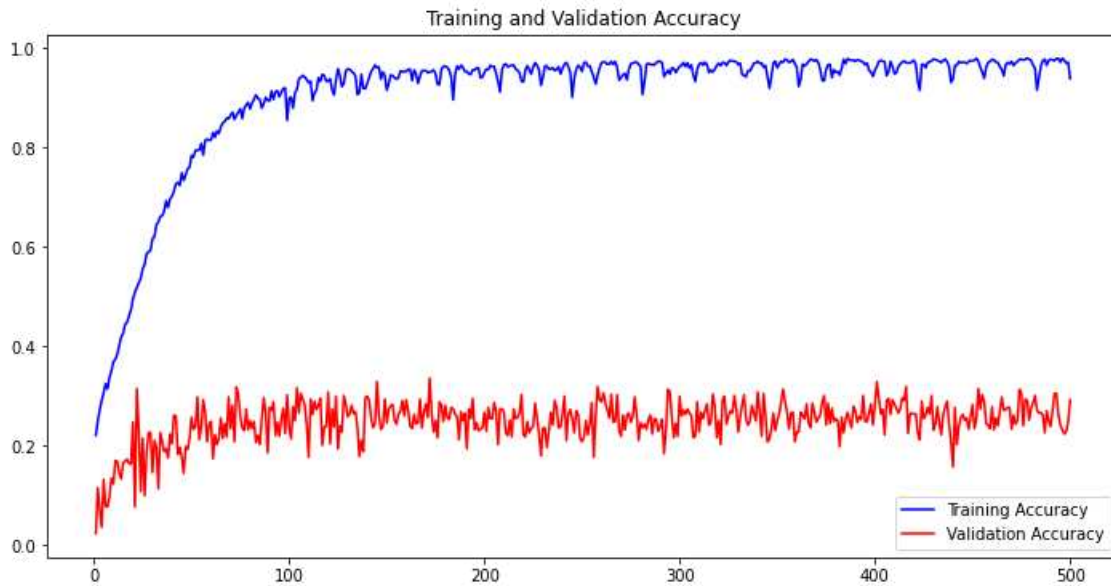


Figure 32: Training and validation accuracy using hold-out cross-validation for implementation 2.

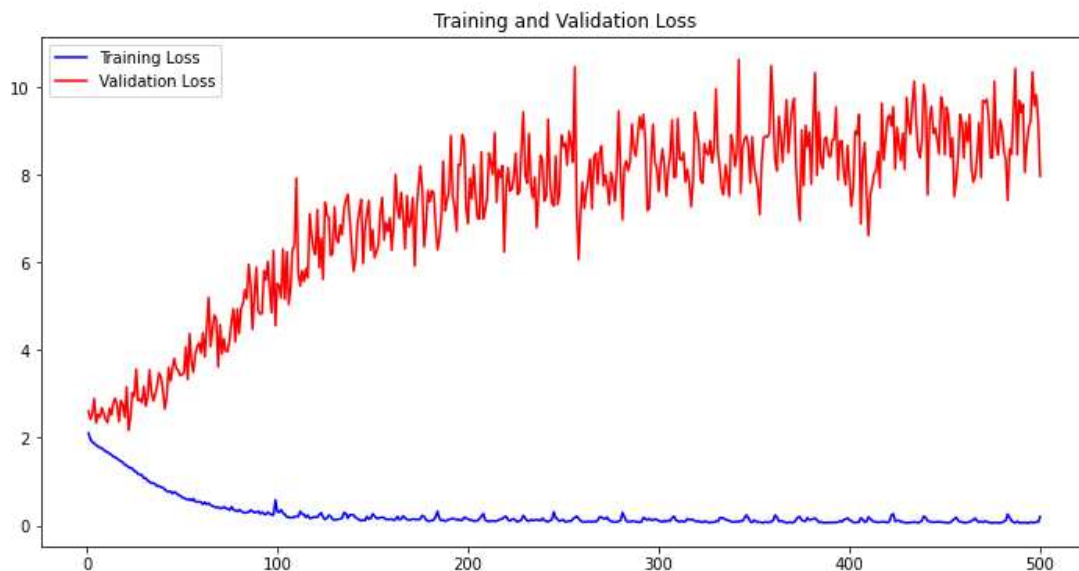


Figure 33: Training and validation loss using hold-out cross-validation for implementation 2.



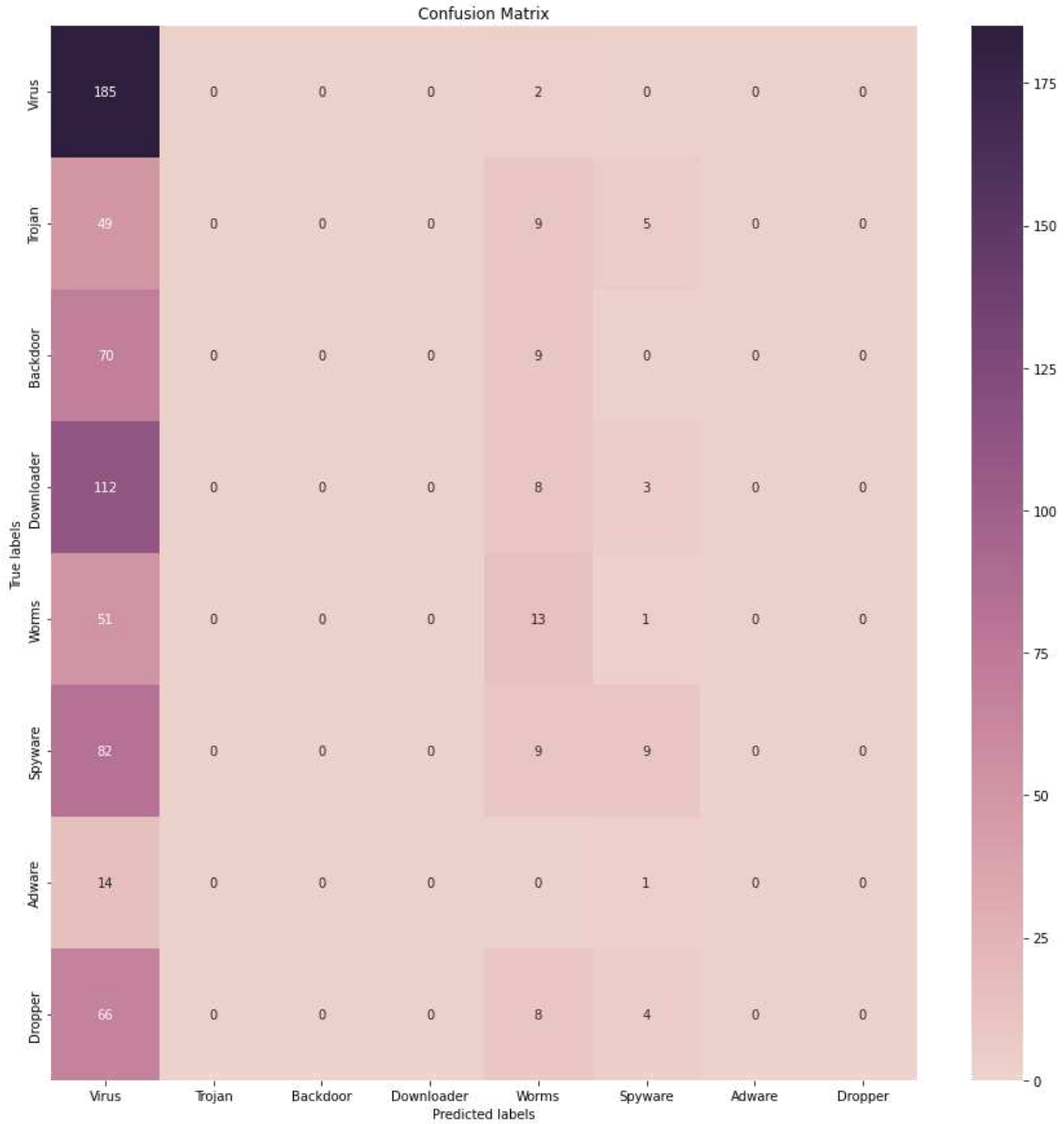


Figure 34: Confusion matrix for Implementation 2

It can be observed from figure 34 how the model misclassified most of the other malware families (Trojan, Backdoor, Downloader, Worms, Spyware, Adware, and Dropper) to be Viruses.

## 4.5 Limitations

- When the model was trained at various times, each time, the training and validation accuracy differed with variances.
- The model performed well in implementation one as the test set was a **subset** of the training dataset, i.e., the model was making predictions based on data it had already seen while being trained. The results are only of value if used in an environment to detect predictable attacks/ already known attacks, but we already have signature-based anti-virus solutions. Implementation 1 is an example of a well-working model but is far from ideal because of the data used to test it. As mentioned, the model is only as good as the data it is fed, i.e., trained on.
- The dataset only works with the eight malware classes, and there is an imbalance observed in the dataset, i.e., Spyware consisted of 832 samples, Adware consisted of 379 samples, and Dropper – 891. Meanwhile, Downloader, Trojan, Worms, Virus, and Backdoor – each consisted of 1001 samples [90].
- Assuming the model works well and is used in real-time, the model trained on this type of dataset may raise false alarms or predictions solely because the model is trained to detect malware by only learning about various classes of malware. It may classify the regular Windows application API calls to be malware too.
- In implementation 2, the model can train well but cannot generalize well, resulting in overfitting. Due to limited time, regularization, or other techniques to solve overfitting could not be implemented.

## 4.6 Key Findings: AI in detection, prevention, analysis, and response to malware

- In the context of prevention, analysis, and response, due to time constraints, AI's feasibility could not be tested.
- Detection of malware is vital to the prevention, analysis and formulation or execution of a response to malware.
- The present application of AI in the industry is in the detection context and is known to do an excellent job. The focus of this research was mainly on the detection aspect of malware. One of the many approaches in AI was tried and tested to detect malware. Although the results were not accurate or ideal, it can still be concluded that AI is feasible for detecting malware.
- AI can assist in the detection and may assist in prevention again by detecting attacks. But there is much more to prevention than just detection, as stated in chapter 3.
- In analysis, AI could assist the threat analyst in studying attacks, uncovering any patterns in attacks, and helping in predicting new attacks, basically assisting in threat hunting. The feasibility of AI may be tested in this context.
- Response to malware largely depends on the human aspect, as humans/organizations usually implement a response. Incident response plans are formulated as a document consisting of several pages describing the steps to be taken in case of a malware attack. Different responses are drafted depending on the type of malware attack. A ransomware attack is treated differently from a phishing attack. The application of AI in this context is infeasible or uncertain.

# Chapter 5: Conclusion & Future Considerations

With the ever-increasing problem of malware and increasing ransom payouts, solutions have been designed to solve this problem. There is no single solution to the malware problem, considering the diverse variants, entry points, and system vulnerabilities. An approach was taken in this project to test the implementation of AI in detecting malware. This is developed to be integrated with existing solutions that could compensate for the existing solutions' disadvantages, like time consumption and predicting new attacks. Although the model implemented in this project has yet to achieve the aspect of predicting new attacks, there are so many other AI models this dataset may work better with and generalize better. This is only one of the many approaches. AI is used in cybersecurity and is known to do an excellent job detecting.

Analyzing artificial intelligence techniques to detect, prevent, analyze, and respond to malware could empower malware detection but cannot prevent, analyze, or respond to malware. It cannot yet replace the human aspect of preventing, analyzing, and responding to malware. It assists in detecting but is not independent of humans in this context.

Various sequential models can be used to implement this dataset in the future. A more updated and larger dataset consisting of the eight malware classes and others could be used to detect other malware families. Training the model on standard or normal behaviour (of Windows application) can help with better performance in real-time as the number of false positives would reduce.

## Future Work

- There are other ways to implement this dataset. The following sequential models can be used.
  - LSTM (Long Short-Term Memory)
  - RNN (Recurrent Neural Network)
- Regularization techniques may solve the problem of overfitting in implementation 2, Chapter 4.
- The dataset can be expanded to eliminate imbalance which may improve the results in Implementation 2, Chapter 4.
- The application of AI can be further researched and tested for the feasibility of its implementation in the context of analysis of malware, prevention, and response to malware.

## Works Cited

- [1] Kaspersky, "A Brief History of Computer Viruses & What the Future Holds," Kaspersky, [Online]. Available: <https://www.kaspersky.com/resource-center/threats/a-brief-history-of-computer-viruses-and-what-the-future-holds>. [Accessed 20 February 2023].
- [2] V. Saengphaibul, "A Brief History of The Evolution of Malware," Fortinet, 15 March 2022. [Online]. Available: <https://www.fortinet.com/blog/threat-research/evolution-of-malware>. [Accessed January 2023].
- [3] Wikipedia, "Brain (computer virus)," Wikipedia, 17 October 2022. [Online]. Available: [https://en.wikipedia.org/wiki/Brain\\_\(computer\\_virus\)](https://en.wikipedia.org/wiki/Brain_(computer_virus)). [Accessed 18 February 2023].
- [4] Radware, "Morris Worm," Radware, [Online]. Available: <https://www.radware.com/security/ddos-knowledge-center/ddospedia/morris-worm/#:~:text=According%20to%20Morris%2C%20the%20purpose,connected%20to%20ARPANET%20in%201988..> [Accessed 20 February 2023].
- [5] E. Root, "ILOVEYOU: the virus that loved everyone," kaspersky, 08 August 2022. [Online]. Available: <https://www.kaspersky.com/blog/cybersecurity-history-iloveyou/45001/>. [Accessed 21 January 2023].
- [6] A. Klein, "Fake Government Attack Reveton Hijacks Computers for Ransom," SecurityIntelligence, 01 May 2012. [Online]. Available: <https://securityintelligence.com/fake-government-attack-reveton-hijacks-computers-ransom/>. [Accessed 21 January 2023].
- [7] B. Nibley, "Bitcoin Price History: 2009 - 2023," SoFi, 14 December 2022. [Online]. Available: <https://www.sofi.com/learn/content/bitcoin-price-history/>. [Accessed 21 January 2023].
- [8] N. Grigorik, "File:CryptoLocker.jpg," Wikimedia Commons, 23 December 2017. [Online]. Available: <https://commons.wikimedia.org/wiki/File:CryptoLocker.jpg>. [Accessed 21 January 2023].
- [9] J. Dossett, "A timeline of the biggest ransomware attacks," CNET, 15 November 2021. [Online]. Available: <https://www.cnet.com/personal-finance/crypto/a-timeline-of-the-biggest-ransomware-attacks/>. [Accessed 21 January 2023].
- [10] K. Baker, "RANSOMWARE AS A SERVICE (RAAS) EXPLAINED HOW IT WORKS & EXAMPLES," CrowdStrike, 30 January 2023. [Online]. Available: <https://www.crowdstrike.com/cybersecurity-101/ransomware/ransomware-as-a-service-raas/>. [Accessed 20 February 2023].
- [1] S. Cook, "2018-2022 Ransomware statistics and facts," comparitech, 6 October 2022. [Online]. Available: <https://www.comparitech.com/antivirus/ransomware-statistics/>. [Accessed 21 January 2023].
- [1] T. Contributor, "passive attack," TechTarget, July 2021. [Online]. Available: <https://www.techtarget.com/whatis/definition/passive-attack>. [Accessed 25 February 2022].
- [1] SQL-Server-Team, "Understanding Server Traffic logs and detecting Denial of Service Attacks," Microsoft, 23 March 2019. [Online]. Available: <https://techcommunity.microsoft.com/t5/sql-server-blog/understanding-server-traffic-logs-and-detecting-denial-of/ba-p/385529>. [Accessed 27 February 2023].

- [1] "What is a denial of service attack (DoS) ?," paloaltonetworks, [Online]. Available:  
4] <https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos>.  
[Accessed 27 February 2023].
- [1] Malwarebytes, "Drive-by download," Malwarebytes, [Online]. Available:  
5] <https://www.malwarebytes.com/glossary/drive-by-download>. [Accessed 22 February 2023].
- [1] A. Levine, "Phishing," Wikipedia, 5 September 2007. [Online]. Available:  
6] <https://commons.wikimedia.org/w/index.php?curid=549747>. [Accessed 27 February 2023].
- [1] A. Magnusson, "What is an Attack Vector? 15 Common Attack Vectors to Know,"  
7] strongdm, 2023 5 January. [Online]. Available: <https://www.strongdm.com/blog/attack-vector>. [Accessed 29 January 2023].
- [1] Microsoft Security, "What is business email compromise (BEC)?," Microsoft, [Online].  
8] Available: [https://www.microsoft.com/en-us/security/business/security-101/what-is-business-email-compromise-bec#:~:text=Business%20email%20compromise%20\(BEC\)%20is%20a%20type%20of%20cybercrime%20where,can%20use%20in%20another%20scam..](https://www.microsoft.com/en-us/security/business/security-101/what-is-business-email-compromise-bec#:~:text=Business%20email%20compromise%20(BEC)%20is%20a%20type%20of%20cybercrime%20where,can%20use%20in%20another%20scam..) [Accessed 29 January 2023].
- [1] A. Sigismondi, "Downtime: The Real Cost Of Ransomware," Delphix, 2021 29 September.  
9] [Online]. Available: <https://www.delphix.com/blog/downtime-real-cost-ransomware>.  
[Accessed 28 January 2023].
- [2] Sophos, "The State of Ransomware 2022," April 2022. [Online]. Available:  
0] <https://assets.sophos.com/X24WTUEQ/at/4zpw59pnkpxnhfhgj9bxgj9/sophos-state-of-ransomware-2022-wp.pdf>. [Accessed 29 January 2023].
- [2] Malwarebytes, "What is a data breach?," Malwarebytes, [Online]. Available:  
1] <https://www.malwarebytes.com/data-breach>. [Accessed 29 January 2023].
- [2] IBM, "Cost of a data breach 2022," July 2022. [Online]. Available:  
2] <https://www.ibm.com/reports/data-breach>. [Accessed 29 January 2023].
- [2] E. Sullivan, "data loss," TechTarget, [Online]. Available:  
3] <https://www.techtarget.com/searchdatabackup/definition/Data-loss>. [Accessed 29 January 2023].
- [2] D. Puglia, "Are Enterprises Ready for Billions of Devices to Join the Internet?," Wired,  
4] [Online]. Available: <https://www.wired.com/insights/2014/12/enterprises-billions-of-devices-internet/>. [Accessed 22 January 2023].
- [2] CrowdStrike, "FILELESS MALWARE EXPLAINED," CrowdStrike, 22 March 2022.  
5] [Online]. Available: <https://www.crowdstrike.com/cybersecurity-101/malware/fileless-malware/>. [Accessed 22 January 2023].
- [2] Malwarebytes, "Computer worm," Malwarebytes, [Online]. Available:  
6] <https://www.malwarebytes.com/computer-worm>. [Accessed 29 January 2023].
- [2] P. Tavares, "Popular evasion techniques in the malware landscape," Infosec, 26 July 2022.  
7] [Online]. Available: <https://resources.infosecinstitute.com/topic/popular-evasion-techniques-in-the-malware-landscape/>. [Accessed 2 February 2023].
- [2] D. K. Marc Ph. Stoecklin co-authored by Jiyong Jang, "DeepLocker: How AI Can Power a  
8] Stealthy New Breed of Malware," SecurityIntelligence, 8 August 2018. [Online]. Available:

- <https://securityintelligence.com/deeplocker-how-ai-can-power-a-stealthy-new-breed-of-malware/>. [Accessed 2 February 2023].
- [2] K. Vanherle, "Overview of AI techniques," Medium, 8 March 2021. [Online]. Available: 9] <https://medium.com/unpackai/overview-of-ai-techniques-d7aeefbc4e20>. [Accessed 11 February 2023].
- [3] R. Anyoha, "The History of Artificial Intelligence," Harvard, 28 August 2017. [Online]. 0] Available: <https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/>. [Accessed 4 February 2023].
- [3] T. E. o. E. Britannica, "Moore's law," Britannica, [Online]. Available: 1] <https://www.britannica.com/technology/Moores-law>. [Accessed 4 February 2023].
- [3] B. Lutkevich, "artificial general intelligence (AGI)," TechTarget, January 2023. [Online]. 2] Available: <https://www.techtarget.com/searchenterpriseai/definition/artificial-general-intelligence-AGI>. [Accessed 9 February 2023].
- [3] IBM, "What is computer vision?," IBM, [Online]. Available: 3] <https://www.ibm.com/topics/computer-vision>. [Accessed 9 February 2023].
- [3] IBM, "What is natural language processing (NLP)?," IBM, [Online]. Available: 4] <https://www.ibm.com/topics/natural-language-processing>. [Accessed 9 February 2023].
- [3] V. S. Bisen, "AI in Robotics: Use of Artificial Intelligence in Robotics," Medium, 14 5] October 2020. [Online]. Available: <https://medium.com/vsinghbisen/ai-in-robotics-use-of-artificial-intelligence-in-robotics-726a4e9ade18>. [Accessed 11 February 2023].
- [3] S. Brown, "Machine learning, explained," MIT Management Sloan School, 21 April 2021. 6] [Online]. Available: <https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained>. [Accessed 12 February 2023].
- [3] A. Ng, "What Artificial Intelligence Can and Can't Do Right Now," Harvard Business 7] Review, 09 November 2016. [Online]. Available: <https://hbr.org/2016/11/what-artificial-intelligence-can-and-cant-do-right-now>. [Accessed 12 February 2023].
- [3] T. W. MALONE, A. & E. V. R. LAUBACHER, D. RUS and M. T. F. , "ARTIFICIAL 8] INTELLIGENCE AND THE FUTURE OF WORK," MIT, 17 December 2020. [Online]. Available: <https://workofthefuture.mit.edu/wp-content/uploads/2020/12/2020-Research-Brief-Malone-Rus-Laubacher2.pdf>. [Accessed 12 February 2023].
- [3] J. Peng, E. C. Jury, P. Dönnies and C. Ciurtin, "Machine Learning Techniques for 9] Personalised Medicine Approaches in Immune-Mediated Chronic Inflammatory Diseases: Applications and Challenges," Frontiers in Pharmacology, 30 September 2021. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fphar.2021.720694/full>. [Accessed 18 February 2023].
- [4] javaTpoint, "Supervised Machine Learning," javaTpoint, [Online]. Available: 0] <https://www.javatpoint.com/supervised-machine-learning>. [Accessed 18 February 2023].
- [4] Google Developers, "Generalization," Google, 18 July 2022. [Online]. Available: 1] <https://developers.google.com/machine-learning/crash-course/generalization/video-lecture>. [Accessed 6 March 2023].
- [4] M. Banoula, "Machine Learning Steps: A Complete Guide!," simplilearn, 16 February 2] 2023. [Online]. Available: <https://www.simplilearn.com/tutorials/machine-learning-tutorial/machine-learning-steps>. [Accessed 18 February 2023].

- [4] B. Li, P. Lu and v.-c. , "Normalize Data component," Microsoft, 11 April 2021. [Online].  
 3] Available: <https://learn.microsoft.com/en-us/azure/machine-learning/component-reference/normalize-data>. [Accessed 18 February 2023].
- [4] S. Kapadia, "6 Steps towards a Successful Machine Learning Project," Medium, 3 August  
 4] 2022. [Online]. Available: <https://towardsdatascience.com/6-steps-towards-a-successful-machine-learning-project-3a56f59e2747>. [Accessed 18 February 2023].
- [4] D. Castillo, "Machine Learning Regression Explained," Seldon, 29 October 2021. [Online].  
 5] Available: <https://www.seldon.io/machine-learning-regression-explained#:~:text=Machine%20Learning%20Regression%20is%20a,used%20to%20predict%20continuous%20outcomes..> [Accessed 18 February 2023].
- [4] R. Gandhi, "Introduction to Machine Learning Algorithms: Linear Regression," Medium, 27  
 6] May 2018. [Online]. Available: <https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a>. [Accessed 20 February 2023].
- [4] Canley, "File:Exam pass logistic curve.svg," Wikipedia, 27 March 2022. [Online].  
 7] Available: [https://commons.wikimedia.org/wiki/File:Exam\\_pass\\_logistic\\_curve.svg](https://commons.wikimedia.org/wiki/File:Exam_pass_logistic_curve.svg). [Accessed 20 February 2023].
- [4] IBM, "What is logistic regression?," IBM, [Online]. Available:  
 8] <https://www.ibm.com/topics/logistic-regression#:~:text=Logistic%20regression%20estimates%20the%20probability,bounded%20between%200%20and%201..> [Accessed 20 February 2023].
- [4] IBM, "What is unsupervised learning?," IBM, [Online]. Available:  
 9] <https://www.ibm.com/topics/unsupervised-learning>. [Accessed 18 February 2023].
- [5] GeeksforGeeks, "Clustering in Machine Learning," GeeksforGeeks, 11 January 2023.  
 0] [Online]. Available: <https://www.geeksforgeeks.org/clustering-in-machine-learning/>. [Accessed 20 February 2023].
- [5] J. M. Carew, "reinforcement learning," techtarget, February 2023. [Online]. Available:  
 1] <https://www.techtarget.com/searchenterpriseai/definition/reinforcement-learning#:~:text=Reinforcement%20learning%20is%20a%20machine,learn%20through%20trial%20and%20error..> [Accessed 18 February 2023].
- [5] Megajuce, "Reinforcement learning," Wikipedia, 10 April 2017. [Online]. Available:  
 2] <https://commons.wikimedia.org/w/index.php?curid=57895741>. [Accessed 26 February 2023].
- [5] Amazon, "What Is A Neural Network?," Amazon, [Online]. Available:  
 3] <https://aws.amazon.com/what-is/neural-network/#:~:text=A%20neural%20network%20is%20a,that%20resembles%20the%20human%20brain..> [Accessed 20 February 2023].
- [5] L. Hardesty, "Explained: Neural networks," MIT, 14 April 2017. [Online]. Available:  
 4] <https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414>. [Accessed 20 February 2023].
- [5] Learn.Genetics, "Neurons Transmit Messages In The Brain," Learn.Genetics, [Online].  
 5] Available: <https://learn.genetics.utah.edu/content/neuroscience/neurons>. [Accessed 20 February 2023].



- [5] E. Kavlakoglu, "AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the Difference?," IBM, 27 May 2020. [Online]. Available: <https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks>. [Accessed 20 February 2023].
- [5] D. Johnson, "Back Propagation in Neural Network: Machine Learning Algorithm," Guru99, 31 December 2022. [Online]. Available: <https://www.guru99.com/backpropagation-neural-network.html>. [Accessed 6 March 2023].
- [5] S. Saha, "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way," Medium, 15 December 2018. [Online]. Available: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. [Accessed 26 February 2023].
- [5] IBM, "Convolutional Neural Networks," IBM, [Online]. Available: <https://www.ibm.com/topics/convolutional-neural-networks>. [Accessed 26 February 2023].
- [6] data school, "Simple guide to confusion matrix terminology," data school, 25 March 2014. [Online]. Available: <https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/>. [Accessed 27 February 2023].
- [6] V. Lyashenko and A. Jha, "Cross-Validation in Machine Learning: How to Do It Right," Neptune, 31 January 2023. [Online]. Available: <https://neptune.ai/blog/cross-validation-in-machine-learning-how-to-do-it-right>. [Accessed 27 February 2023].
- [6] A. SHRIVASTAVA, "Underfitting Vs Just right Vs Overfitting in Machine learning," Kaggle, 2020. [Online]. Available: <https://www.kaggle.com/getting-started/166897>. [Accessed 27 February 2023].
- [6] N. Duggal, "Advantages and Disadvantages of Artificial Intelligence," simplilearn, 13 February 2023. [Online]. Available: <https://www.simplilearn.com/advantages-and-disadvantages-of-artificial-intelligence-article>. [Accessed 19 February 2023].
- [6] S. Sheridan, "AI for Manufacturing: Why You Need to Digitize Your Manufacturing Process," Levity, 16 November 2022. [Online]. Available: <https://levity.ai/blog/ai-for-manufacturing-why-you-need-to-digitize-your-manufacturing-process#:~:text=Manufacturers%20can%20save%20thousands%20of,the%20need%20for%20human%20intervention..> [Accessed 19 February 2023].
- [6] IBM, "What is artificial intelligence in medicine?," IBM, [Online]. Available: <https://www.ibm.com/topics/artificial-intelligence-medicine#:~:text=How%20is%20artificial%20intelligence%20used,health%20outcomes%20and%20patient%20experiences..> [Accessed 19 February 2023].
- [6] 10xDS Team, "Top 10 Benefits of Artificial Intelligence (AI)," 10xDS, 30 August 2020. [Online]. Available: <https://10xds.com/blog/benefits-of-artificial-intelligence-ai/>. [Accessed 19 February 2023].
- [6] Merriam-Webster.com Dictionary, "recidivism," [Online]. Available: <https://www.merriam-webster.com/dictionary/recidivism>. [Accessed 12 February 2023].
- [6] Logically, "5 Examples of Biased Artificial Intelligence," 30 July 2019. [Online]. Available: <https://www.logically.ai/articles/5-examples-of-biased-ai>. [Accessed 12 February 2023].
- [6] C. Arena, "7 Disadvantages of Artificial Intelligence Everyone Should Know About," Liberties, 14 June 2022. [Online]. Available:

- <https://www.liberties.eu/en/stories/disadvantages-of-artificial-intelligence/44289>. [Accessed 19 February 2023].
- [7 I. Sample, "What are deepfakes – and how can you spot them?," The Guardian, 13 January 0] 2020. [Online]. Available: <https://www.theguardian.com/technology/2020/jan/13/what-are-deepfakes-and-how-can-you-spot-them>. [Accessed 19 February 2023].
- [7 C. W. Ron Schmelzer, "Towards A More Transparent AI," Forbes, 23 May 2020. [Online]. 1] Available: <https://www.forbes.com/sites/cognitiveworld/2020/05/23/towards-a-more-transparent-ai/?sh=29075f683d93>. [Accessed 19 February 2023].
- [7 S. D. Gantz and D. R. Philpott, "Malware Detection," ScienceDirect, 2013. [Online]. 2] Available: <https://www.sciencedirect.com/topics/computer-science/malware-detection#:~:text=Among%20the%20more%20familiar%20forms,other%20forms%20of%20malicious%20code..> [Accessed 9 February 2023].
- [7 Ö. ASLAN and R. SAMET , "A Comprehensive Review on Malware Detection 3] Approaches," 3 January 2020. [Online]. Available: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8949524>. [Accessed 19 February 2023].
- [7 CrowdStrike, "WHAT IS A POLYMORPHIC VIRUS? DETECTION AND BEST 4] PRACTICES," CrowdStrike, 22 July 2022. [Online]. Available: <https://www.crowdstrike.com/cybersecurity-101/malware/polymorphic-virus/#:~:text=A%20polymorphic%20virus%2C%20sometimes%20referred,files%20through%20new%20decryption%20routines..> [Accessed 19 February 2023].
- [7 R. Awati, "metamorphic and polymorphic malware," Techtarget, March 2022. [Online]. 5] Available: <https://www.techtarget.com/searchsecurity/definition/metamorphic-and-polymorphic-malware>. [Accessed 19 February 2023].
- [7 R. Awati, "stealth virus," Techtarget, December 2021. [Online]. Available: 6] <https://www.techtarget.com/searchsecurity/definition/stealth-virus>. [Accessed 19 February 2023].
- [7 ExtraHop, "MALWARE OBFUSCATION: TECHNIQUES, DEFINITION & 7] DETECTION," ExtraHop, [Online]. Available: <https://www.extrahop.com/resources/attacks/malware-obfuscation/#:~:text=Malware%20obfuscation%20is%20the%20act,to%20hide%20its%20presence%20completely..> [Accessed 19 February 2023].
- [7 M. Sikorski and A. Honig, "Packed and Obfuscated Malware," O'Reilly, [Online]. 8] Available: <https://www.oreilly.com/library/view/practical-malware-analysis/9781593272906/ch02s04.html>. [Accessed 19 February 2023].
- [7 Z. Bazrafshan, H. Hashemi, S. M. H. Fard and A. Hamzeh, "A survey on heuristic malware 9] detection techniques," in *The 5th Conference on Information and Knowledge Technology*, Shiraz, Iran, 2013.
- [8 T. Taylor, "Behavior-Based Security Vs. Signature-Based Security: How They Differ," 0] TechGenix, 26 November 2019. [Online]. Available: <https://techgenix.com/behavior-based-security/>. [Accessed 20 February 2023].
- [8 A. Shabtai, R. Moskovitch, C. Feher, S. Dolev and Y. Elovici, "Detecting unknown 1] malicious code by applying classification techniques on OpCode patterns," 27 February 2012. [Online]. Available: [75](https://security-</a></p></div><div data-bbox=)

- informatics.springeropen.com/articles/10.1186/2190-8532-1-1#citeas. [Accessed 20 February 2023].
- [8 M. Souppaya and K. Scarfone, "Guide to Malware Incident Prevention and Handling for  
2] Desktops and Laptops," July 2013. [Online]. Available:  
https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-83r1.pdf. [Accessed 19  
February 2023].
- [8 VMware, "What is an intrusion prevention system?," VMware, [Online]. Available:  
3] https://www.vmware.com/topics/glossary/content/intrusion-prevention-  
system.html#:~:text=What%20is%20an%20intrusion%20prevention,it%2C%20when%20it  
%20does%20occur.. [Accessed 20 February 2023].
- [8 paloaltonetworks, "What is a Zero Trust Architecture," paloaltonetworks, [Online].  
4] Available: https://www.paloaltonetworks.com/cyberpedia/what-is-a-zero-trust-architecture.  
[Accessed 20 February 2023].
- [8 VMware, "What is Malware Analysis?," VMware, [Online]. Available:  
5] https://www.vmware.com/topics/glossary/content/malware-  
analysis.html#:~:text=Malware%20Analysis%20is%20the%20practice,analysis%2C%20or  
%20full%20reverse%20engineering.. [Accessed 20 February 2023].
- [8 M. H. W. D.Sc., "mnemonic code," 30 November 2017. [Online]. Available:  
6] https://link.springer.com/referenceworkentry/10.1007/1-4020-0613-  
6\_11624#:~:text=A%20code%20that%20can%20be,recalling%20the%20information%20it  
%20represents.. [Accessed 20 February 2023].
- [8 K. Johnson, "Why is malware analysis important?," TechTarget, December 2022. [Online].  
7] Available: https://www.techtarget.com/searchsecurity/feature/Why-is-malware-analysis-  
important. [Accessed 27 February 2023].
- [8 L. Kessem and M. Mayne, "Definitive guide to ransomware 2022," May 2022. [Online].  
8] Available: https://www.ibm.com/downloads/cas/EV6NAQR4. [Accessed 27 February  
2023].
- [8 A. Walker, "What is an API? Full Form, Meaning, Definition, Types & Example," Guru99,  
9] 24 December 2022. [Online]. Available: https://www.guru99.com/what-is-api.html.  
[Accessed 18 February 2023].
- [9 F. O. Catak, "ocatak/malware\_api\_class," 22 November 2021. [Online]. Available:  
0] https://github.com/ocatak/malware\_api\_class. [Accessed 5 February 2023].
- [9 A. D'Agostino, "Get started with TensorFlow 2.0 — Introduction to deep learning,"  
1] Medium, 22 November 2022. [Online]. Available: https://towardsdatascience.com/a-  
comprehensive-introduction-to-tensorflows-sequential-api-and-model-for-deep-learning-  
c5e31aee49fa#:~:text=The%20sequential%20model%20allows%20us,for%20building%20d  
eep%20learning%20models.. [Accessed 5 February 2023].
- [9 A. Ng, J. Ngiam, C. Y. Foo, Y. Mai, C. Suen, A. Coates, A. Maas, A. Hannun, B. Huval, T.  
2] Wang and S. Tandon, "Softmax Regression," Stanford, [Online]. Available:  
http://deeplearning.stanford.edu/tutorial/supervised/SoftmaxRegression/. [Accessed 4  
February 2023].
- [9 Google, "Multi-Class Neural Networks: Softmax," Google, 18 July 2022. [Online].  
3] Available: https://developers.google.com/machine-learning/crash-course/multi-class-neural-  
networks/softmax. [Accessed 5 February 2023].

- [9 4] A. D'Agostino, "Introduction to neural networks — weights, biases and activation," Medium, 27 December 2021. [Online]. Available: <https://medium.com/mllearning-ai/introduction-to-neural-networks-weights-biases-and-activation-270ebf2545aa>. [Accessed 5 February 2023].
- [9 5] C. Pere, "What is activation function ?," Medium, 9 June 2020. [Online]. Available: <https://towardsdatascience.com/what-is-activation-function-1464a629cdca>. [Accessed 5 February 2023].
- [9 6] J. Brownlee, "Your First Deep Learning Project in Python with Keras Step-by-Step," Machine Learning Mastery, 18 June 2022. [Online]. Available: <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/>. [Accessed 25 February 2023].
- [9 7] Google, "Classification: Accuracy," Google, 18 July 2022. [Online]. Available: <https://developers.google.com/machine-learning/crash-course/classification/accuracy#:~:text=Accuracy%20is%20one%20metric%20for,prediction s%20Total%20number%20of%20predictions>. [Accessed 6 February 2023].
- [9 8] Bharathi, "Confusion Matrix for Multi-Class Classification," Analytics Vidhya, 24 June 2021. [Online]. Available: <https://www.analyticsvidhya.com/blog/2021/06/confusion-matrix-for-multi-class-classification/>. [Accessed 27 February 2023].
- [9 9] H. Férée, "File:The Imitation Game.svg," Wikimedia, 21 October 2011. [Online]. Available: [https://commons.wikimedia.org/wiki/File:The\\_Imitation\\_Game.svg](https://commons.wikimedia.org/wiki/File:The_Imitation_Game.svg). [Accessed 4 February 2023].
- [1 00] J. A. S. Margallo, "File:Turing test diagram.png," Wikimedia, 22 March 2017. [Online]. Available: [https://commons.wikimedia.org/wiki/File:Turing\\_test\\_diagram.png](https://commons.wikimedia.org/wiki/File:Turing_test_diagram.png). [Accessed 4 February 2023].
- [1 01] A. Gonfalonieri, "Introduction to Causality in Machine Learning," Medium, 8 July 2020. [Online]. Available: <https://towardsdatascience.com/introduction-to-causality-in-machine-learning-4cee9467f06fity-in-machine-learning-4cee9467f06f>. [Accessed 18 February 2023].