ALGORITHM DEVELOPMENT FOR IMPROVING CYBER THREAT INTELLIGENCE AGAINST ONION SERVICE

By

MUHAMMAD FAIZAN RAJA

NATIONAL UNIVERSITY OF MODERN LANGUAGES

ISLAMABAD

December, 2024

ALGORITHM DEVELOPMENT FOR IMPROVING CYBER THREAT INTELLIGENCE AGAINST ONION SERVICE

By

Muhammad Faizan Raja
BSSE, Virtual University of Pakistan, Lahore, 2021

A THESIS SUBMITTED IN PARTIAL

FULFILLMENT OFTHE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

In **Software Engineering**

To

FACULTY OF ENGINEERING & COMPUTING

NATIONAL UNIVERSITY OF MODERN LANGUAGES ISLAMABAD

THESIS AND DEFENSE APPROVAL FORM

The undersigned certify that they have read the following thesis, examined the defense, are satisfied with overall exam performance, and recommend the thesis to the Faculty of Engineering andComputer Sciences for acceptance.

Thesis Title: ALGORITHM DEVELOPMENT FOR IMPROVING CYBER THREAT INTELLIGENCE AGAINST ONION SERVICE

Submitted by: Muhammad Faizan Raja **Registration #:** 57 MS/SE/S22

Master of Science in Software Engineering
Degree name in full

Software Engineering
Discipline

Dr. Muzafar Khan _____
Research Supervisor Signature of Research Supervisor

Dr. Raheel Zafar _____
Research Co-Supervisor Signature of Co- Supervisor

Dr. Sumaira Nazir _____
HOD (SE) Signature of HOD (SE)

Dr. Muhammad Noman Malik _____
Dean (FE&C) Signature of Dean (FE&C)

December 09th, 2024

AUTHOR'S DECLARATION

I Muhammad Faizan Raja

Son of Raja Muhammad Rafique

Registration # 57 MS/SE/S22

Discipline Software Engineering

Candidate of **Master of Science in Software Engineering (MSSE)** at the National University of Modern Languages do hereby declare that the thesis **ALGORITHM DEVELOPMENT FOR IMPROVING CYBER THREAT INTELLIGENCE AGAINST ONION SERVICE** submitted by me in partial fulfillment of MSSE degree is my original work, and has not been submitted or published earlier. I also solemnly declare that it shall not, in the future, be submitted by me for obtaining any other degree from this or any other university or institution. I also understand that if evidence of plagiarism is found in my thesis/dissertation at any stage, even afterthe award of a degree, the work may be canceled and the degree revoked.

_____

Signature of Candidate

Muhammad Faizan Raja

Name of Candidate

December, 2024

Date

ABSTRACT

ALGORITHM DEVELOPMENT FOR IMPROVING CYBERTHREAT INTELLIGENCE AGAINST ONION SERVICE

Web applications are widely used in various business domains due to their affordability andplatform independence. Billions of users rely on these applications to perform daily tasks. Cybersecurity is safeguarding computer systems, networks, and data from unauthorized access, theft, and other malicious attacks. The Tor network allows users to host anonymous websites known as onion services, or torch hidden services. This helps to conceal the IP addresses of the server and the client, making it difficult for third parties to intercept or monitor the conversation. There is a noticeable rise in the variety of onion services with illicit and criminal intent on the darkweb. In recent years, attackers spread malware through images by embedding malicious code, tricking users into downloading harmful files, and exploiting image-related vulnerabilities. Identifying and neutralizing these threats involves various strategies, but automated malware generation techniques continue to produce malware that resists current detection technologies. It presents unique challenges for cyber threat intelligence, including the increasing difficulty of cyberthreats targeting them. There is a greater need for sophisticated and precise malware classification and detection methods. This study presents a novel approach to malware image classification by employing a convolutional neural network. The results demonstrate a significant accuracy of 96%on the malware image collection. Various metrics such as precision, recall, specificity, and F1 score were employed to evaluate the model's performance. The experiment findings demonstrate the effectiveness of the suggested approach as a reliable technique for detecting malware through images. It enhances detection, analysis, and attribution capabilities, optimizing resource allocation and informing effective mitigation strategies for complex threats within the Tor network. The studyillustrates how deep learning frameworks can reduce the likelihood of malware attacks.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

CTI	Cyber Threat Intelligence
OSINT	OpeN Source Intelligence
Tor	The Onion Routing
DNS	Domain Name System
ICAMM	Internet Corporation for Assigned Names and Numbers
IETF	Internet Engineering Task Force
IoT	Internet of Things
CSA	Cyber Security Awareness
DDoS	Distributed Denial of Service
IIoT	Industrial Internet of Things
ROT	Return-Oriented Programming
ISTR	Internet Security Threat Report
BCSA	Binary Crow Search Algorithm
DNN	Deep Neural Network
HSDirs	Hidden Service Directories
NAT	Network Address Translator
IoCs	Indicators of Compromise
SECDFAN	SECurity Discussion Forums Analysis
Grad-CAM	Gradient-weighted Class Activation Mapping
GIWRF	Gini Impurity-based Weighted Random Forest
CSC	Cyber Supply Chain
CNN	Convolutional Neural Network
AUC	Area Under Curve
MAE	Mean Absolute Error
MSE	Mean Squared Error

LIST OF TABLES

ACKNOWLEDGEMENT

First and foremost, I would like to express my heartfelt gratitude and deep appreciation to Almighty Allah, whose blessings made this study possible and successful. Without divine support,this achievement would not have been possible.

I am immensely thankful to all the individuals and sources whose unwavering support andencouragement played a pivotal role in the completion of this study. Their honest espousal has beeninvaluable, and I am sincerely grateful for their contributions. I owe a special debt of gratitude to myresearch supervisor Dr. Muzafar Khan and Co-Supervisor Dr. Raheel Zafar, whose dedication and guidance were instrumental in shaping my research journey. Their commitment and relentless effortsleft no stone unturned, and I am truly grateful for their mentorship.

To every person who has contributed to my success in ways both seen and unseen, I extendmy heartfelt thanks. Your support has been an indispensable part of this endeavor, and I am deeplyappreciative of everything you have done.

DEDICATION

This thesis serves as a testament to the enduring influence of my family, the bedrock of my aspirations. Gratitude fills my heart as I acknowledge my parents, whose unwavering support andcountless sacrifices have illuminated my academic path. Their belief in my potential has been the propulsive force guiding me through the challenges of academia. To my siblings, whose unwavering encouragement has been a perpetual source of motivation, I express heartfelt appreciation.

In dedicating this work, I extend profound gratitude to my friends and mentors, whose guidance and camaraderie have been indispensable. Their insights and shared experiences have enriched my academic journey, molded not only my intellectual growth but also contributed to theultimate realization of this research endeavor. To those who believed in my capabilities and offeredencouragement, your influence has been truly transformative.

As this thesis takes its place on the academic stage, I dedicate it to the cherished individualswho have played pivotal roles in my life. May this dedication stand as a humble acknowledgmentof the profound impact each of you has had in shaping my academic pursuits. Additionally, I extend appreciation to the broader academic community, whose collective wisdom and resources have been instrumental in the development and completion of this research.

CHAPTER 1

INTRODUCTION

This chapter gives an overview of the research topic, research questions, and their objectives. It outlines the contributions and significance of the study and the context within whichthe research is situated. This sets the stage for the detailed exploration of the subject matter and highlights the broader implications of the research.

1.1 Context

Human civilization has transitioned into the information age due to the expansion of the Internet and other cyberspace technologies. These technologies now cater to everyday needs, including social media, transportation, and utilities. Because of the accessibility of dependable infrastructures like cloud storage, affordable platforms, and a sizable target market, the majority ofenterprises moved online. Nevertheless, several cyber threats, including malware, spam, phishing,financial fraud, etc., can be found on the Internet. Although there are many advantages to these developments, they are also susceptible to malevolent exploitation because everyone relies on them. A malicious website contains potentially hazardous content, such as malware or phishing assaults that infect users' smart devices with malware without requiring them to engage with the website through clicks or downloads. To identify and neutralize cyber risks, the area of Cyber Threat Intelligence (CTI) employs data-driven methods and methodical analysis of large files, binary events, and Open-Source Intelligence (OSINT). Artificial intelligence (AI) has advanced recently to the point that most cyber threats can now be identified with little assistance from humans, improving cybersecurity and lowering the possibility of human error [1]. Even with the advances in cyber security, threats that originate from and persist within the dark web continue to provide persistent issues. These difficulties result from the intricacy of the technology involved as well as the lack of extensive study and investigation into the interpretability of AI models used to assess dangers on the darkweb [2,3].

1.2 Problem Statement

The onion service environment presents unique challenges for cyber threat intelligence, including the anonymized nature of the services and the increasing difficulty of cyber threats targeting them [4]. A malware detection system should be created by looking into the potential of static features utilizing function engineering techniques to improve feature identification and moreprecisely identify malware using deep learning algorithms [5]. The accuracy of cyber threat intelligence in the area of cyber security still has to be improved.

1.3 Research Questions

The research questions are as follows.

RQ 1: What are the specific techniques to improve the cyber threat intelligence to detect malware attacks in the onion service?

RQ 2: How accurate is the algorithm developed for detecting malware attacks in the onionservice?

1.4 Research Objectives

The research objectives are as follows.

Obj 1: To identify the various techniques that can improve the accuracy of cyber threatsto detect malware attacks.

Obj 2: To evaluate the accuracy of the algorithm in identifying different types of cyberthreats in Onion services.

1.5 Motivation

The motivation behind algorithm development for improving cyber threat intelligence against Onion services stems from the increasing complexity and sophistication of cyber threats within the dark web. Onion services, facilitated by networks like Tor, provide anonymity and encryption, making them a preferred platform for illicit activities such as cybercrime, terrorism, and trafficking of illegal goods and services. In recent years, attackers used malicious images to spread malware by embedding code within images, tricking users into downloading harmful files,and exploiting image-related vulnerabilities. Attackers can disguise malware as image files by using double extensions (e.g., photo.jpg.exe) or by embedding malicious code within an

image filethat exploits certain vulnerabilities in the image processing software. Users might be tricked into executing these files, thinking they are harmless images. By embedding malicious code within images or using images in unconventional ways, attackers can evade traditional security mechanisms that rely on signature-based detection. This obfuscation makes it harder for antivirus software to detect the malware. Various strategies exist to identify and neutralize such threats, butmodern automated malware generation techniques keep producing malware that resists current detection technologies. Traditional cyber security measures often struggle to detect these threats

effectively due to the inherent challenges posed by the dark web's anonymity and encryption layers.To address this pressing issue, the development of advanced algorithms for cyber threat intelligenceis crucial. These algorithms are designed to enhance the detection, analysis, and response capabilities of cybersecurity professionals and organizations operating in this space.

1.6 Scope of Study

The study aims to develop advanced algorithms to improve cyber threat intelligence for Onion services, focusing on a range of objectives, methodologies, and outcomes. It involves the comprehensive collection, processing, and analysis of data to create these sophisticated algorithms. By leveraging advanced techniques like machine learning, deep learning, and statistical analysis, the study seeks to enhance the detection, monitoring, and reporting of cyber threats within these obscured networks. The use of machine learning enables the model to learn from large datasets andimprove over time. Deep learning, a subset of machine learning, provides the ability to identify complex patterns and anomalies that traditional methods might miss. Statistical analysis helps in understanding the data distributions and relationships, aiding in the accuracy of threat detection. The ultimate goal is to bolster the safety and security of the digital environment, making it more resilient against cyber-attacks. This research not only aims to address current threats but also anticipates future challenges, ensuring that the solutions remain robust and effective. By focusing on Onion services, which are known for their anonymity and encryption, the study tackles some of the most challenging aspects of cybersecurity. The outcomes of this research are expected to provide valuable insights and tools for cybersecurity professionals, helping them protect sensitiveinformation and maintain the integrity of networks.

1.7 Contribution and Significance.

This research aims to advance cybersecurity strategies and solutions for Onion services byenhancing the accuracy of existing models and exploring new techniques. It seeks to provide practical guidance and recommendations to cybersecurity professionals and organizations. The goal is to optimize cyber threat intelligence capabilities effectively, ensuring robust protection against emerging threats. By refining current methodologies and investigating innovative approaches, this study will contribute to the field of cybersecurity. Ultimately, it aspires to empower organizations with improved tools and strategies, fostering a safer digital environment for Onion services and beyond.

1.8 Thesis Structure

Providing an overview of the internet, its layers, attacks, viruses, forms of attacks, the Tor browser, onion services, cyber threat intelligence, and a review of prior research-based predictions,**Chapter 1** serves as an introduction. It describes the problem statement, the extent of the thesis, and the foundation for accuracy in cyber threat intelligence, and lays the groundwork for the following chapters. The themes mentioned in Chapter 1 are expanded upon in **Chapter 2**. In the literature review, the most important literature is analyzed to find out what has already been studiedand where new research can be done. In **Chapter 3**, important concepts and methodologies are integrated into a logical framework that will be used for the entire study. This chapter includes information about the dataset, the methodology, the results, and the tools and technologies utilizedto produce the desired results. A thorough examination of the study's findings is provided in **Chapter 4**, which also covers the analysis techniques used and highlights important discoveries and results. The research findings and their ramifications are thoroughly explained in this chapter.The thesis is concluded in **Chapter 5**, which provides a summary of the key discoveries made during the investigation and recommendations for further research. It presents an analysis of the study process and suggests additional areas of investigation.

1.9 Summary

The chapter outlines the problem statement, addressing the limitations of CTI accuracy andthe necessity for improved strategies. It poses research questions on enhancing cyber threat intelligence. The research objectives aim to develop advanced algorithms. Motivated by the increasing sophistication of cyber threats in hidden networks, the study's scope encompasses data collection, analysis, and the development of predictive models. The thesis structure includes subsequent chapters expanding on concepts introduced, integrating methodologies, presenting findings, and concluding with recommendations for future research.

CHAPTER 2

LITERATURE REVIEW

This chapter will provide an in-depth review of the background relevant to the research study, exploring existing literature on the internet, its layers, cybersecurity, various cyber-attacks, types of malwares, Onion Services, Tor browser, and related technologies used in detecting malwareattacks.

2.1 Internet

The Internet is a global network of interconnected computer systems that exchange data using the Internet protocol suite (TCP/IP). This extensive network spans local to global scales andconsists of networks from the public, corporate, academic, government, and private sectors. Severalnetworking technologies, including electrical, wireless, and optical connections, link different networks. The Internet offers various information resources and services, such as email, phone calls, file sharing, and interconnected hypertext sites and applications through the World Wide Web(WWW).

In the early 1990s, the National Science Foundation Network (NSFNET) was established as a new backbone, with private financing supporting commercial extensions. This encouraged thedevelopment of new networking technologies and the integration of numerous networks using DARPA's Internet protocol suite. This era also saw the emergence of the World Wide Web and theconnecting of business networks and organizations, marking the arrival of the modern Internet.

The exponential growth of personal, institutional, and mobile computers connected to thenetwork marked a significant era. During the 1980s, the Internet was primarily used by academicsfor research and communication. Over time, the technologies and services associated with the Internet became commercialized. This commercialization transformed the Internet into a vital component of modern life, impacting almost every aspect of daily activities, from communicationand education to business and entertainment. The widespread adoption and integration of Internet technologies have made them essential tools for individuals and

organizations alike. [6].

The Internet has changed and replaced many traditional communication channels like phone, radio, television, mail, and newspapers. New services such as email, Internet phone, InternetTV, online music, digital newspapers, and video streaming websites have become more popular. Books, newspapers, and other printed media have adapted by becoming web feeds, blogs, and onlinenews aggregators, or by integrating website technologies.

Social networking sites and online forums are examples of new kinds of interpersonal communication that have been made possible and expedited by instant messaging. Major merchants, small businesses, and entrepreneurs have come to love Internet shopping. It allows themto operate purely online or grow their physical presence to attract a wider audience. Furthermore, supply chains throughout whole industries have been greatly impacted by internet banking and business-to-business services.

Since no single body is in charge of overseeing the application of technology or setting rules for access and usage, each network component on the Internet establishes its policies. The two main namespaces on the Internet are the Domain Name System (DNS) and the Internet Protocol(IP) address space, which are overseen by the Internet Corporation for Assigned Names and Numbers (ICANN). The technical underpinnings and standardization of the core protocols are overseen by the nonprofit Internet Engineering Task Force (IETF), which is composed of internationally dispersed contributors. The IETF's efforts are open to participation by anyone withtechnical expertise [7].

2.1.1 Internet Layers

The internet is structured into distinct layers, each characterized by its accessibility and functionality.

Surface Web

The surface web refers to the part of the internet that is readily accessible to everyone and can be easily searched using common web search engines like Google or Bing. It includes all the websites and pages that are indexed by these search engines, which means they are organized andlisted in search results when users perform queries.

However, the surface web only represents about 7% of the total content available online. The remaining 93% of the internet consists of areas that are not indexed by search engines.

These areas include private databases, subscription-only content, and other resources that require specialaccess or are not intended for public search. Essentially, while the surface web offers a broad rangeof information, it is only a small fraction of the entire digital landscape [8].

Deep Web

The "deep web" refers to the part of the internet that traditional search engines don't index. To access deep websites, you typically need to use direct URLs or IP addresses. Additionally, some

deep websites may require passwords or other security credentials to view their content.[9]. Deep websites have multiple uses, including webmail, online banking, cloud storage, social media pages and profiles with restricted access, and web forums where content must be registered to see. The deep web also includes some online newspapers and magazines and subscription-based services like video on demand [10].

Dark Web

The term "dark web" refers to Internet content accessible through overlay networks, or "darknets," requiring specific software, configurations, or authorization. Private computer networks can conduct anonymous exchanges and transactions via the dark web without disclosingpersonally identifiable information, like a user's location. It's crucial to remember that, even thoughthe term "deep web" is occasionally misapplied to refer only to the dark web, the deep web refers to the portion of the internet that search engines have not yet indexed [11]. The dark web consists of a variety of private and public organizations that govern darknets. These darknets can range fromsmall peer-to-peer networks to well-known ones like Freenet, Tor, I2P, and Riffle. The traditionalinternet is also known as the Clearnet because it lacks encryption unlike the dark web, which is encrypted. The Tor dark web, also known as onion land, operates under the network's top-level domain suffix .onion, using onion routing traffic anonymization technology.

Figure 2.1: Layers of Internet [7]

2.2 Internet Security

Information technology security (IT security), also referred to as computer security, cyber security, or digital security, involves protecting computer systems and networks from unauthorizedaccess and malicious attacks. The goal is to prevent theft or damage to hardware, software, and data, as well as the exposure of confidential information and disruption of services [12]. The importance of this topic is underscored by the growing reliance on computers, the Internet, and wireless network technologies like Bluetooth and Wi-Fi. This significance is further augmented bythe widespread use of smart devices, such as smartphones, televisions, and other gadgets that constitute the Internet of Things (IoT). Cybersecurity is a major concern in the modern world due to the intricate nature of information systems and their impact on society. Vital systems like powerdistribution, elections, and finance, which oversee extensive networks with significant physical implications, require heightened security measures [13].

Untargeted cyberattacks present a significant risk to everyday internet users because they are not aimed at any specific individual or organization. Instead, attackers cast a wide net, indiscriminately targeting a large number of devices, services, or users. They exploit the inherent openness of the Internet, which allows them to reach a broad audience without needing to tailor their attack to particular targets.

This type of attack can involve methods such as sending malware through email spam, exploiting security vulnerabilities in commonly used software, or using bots to perform distributeddenial-of-service (DDoS) attacks. The goal is to compromise as many systems as possible, often leading to widespread damage or data breaches. Because the attack is not directed at any specific entity, it increases the likelihood of impacting a large number of people and systems, making it a significant threat to general internet security.

Chaudhary et al. [14] address the need for effective metrics to evaluate the effectiveness ofcybersecurity awareness (CSA) programs. Despite organizational investments in CSA, there is often a lack of evaluation, leading to a limited understanding of program efficacy. This knowledgegap may result in wasted resources and a false sense of security. The goal is to assist organizations in improving their CSA programs and, consequently, enhancing their overall cybersecurity posture.They discussed existing literature that included simulated attacks as a method to evaluate CSA

program effectiveness. The results provide valuable insights for organizations seeking to enhancetheir CSA programs.

2.3 Cyber Attack

A cyber-attack, also known as a cyberattack, encompasses any offensive action aimed at computer networks, infrastructures, personal computers, cell phones, or computer information systems. An attacker, whether an individual or entity, seeks unauthorized access to data, features, or restricted areas of the system, often with malicious intent. Depending on the circumstances, cyberattacks may be classified as cyberterrorism or cyberwarfare. These attacks can be initiated byanonymous sources and may involve sovereign states, individuals, groups, societies, or organizations. Occasionally, the term "cyber-weapon" is used to describe tools aiding in cyberattacks. In recent years, there has been a notable increase in cyberattacks, with distributed denial of service (DDoS) attacks being a prominent example [15].

A cyberattack can occur by gaining unauthorized access to a private network or another vulnerable system, allowing the attacker to exploit, alter, or potentially destroy a specific target. There are various types of cyberattacks, ranging from installing spyware on a user's computer to attempting to disrupt an entire country's infrastructure. To differentiate the word from more commonplace data breaches and more widespread hacking activities, legal experts are working torestrict its use to situations in which physical harm is sustained.

2.3.1 Types of Cyber Attacks

In cybersecurity, attacks are often classified into two main types: active attacks and passiveattacks, each with distinct characteristics and impacts on system resources.

An active attack involves actions that modify or disrupt the normal operation of system resources. The attacker actively interferes with the system to alter, damage, or manipulate data or services. The primary goal of an active attack is to cause immediate and noticeable disruptions or harm, impacting the integrity, availability, or functionality of the system [15].

A passive attack, on the other hand, focuses on accessing or obtaining system informationwithout affecting the system's normal operation. This type of attack is stealthier and less likely to be detected, as it does not alter data or services but simply monitors or collects information.

The primary goal of a passive attack is to gather sensitive information or intelligence whileremaining undetected, which can later be used for further attacks or unauthorized access [15]. An attack can be committed by an insider or from outside the organization.

An "inside attack" refers to an attack initiated by an individual within the security perimeter,known as an "insider." This means that the individual has authorized access to system resources but is using them in a manner that was not permitted by the authorizing party.

An "outside attack" is initiated by an unauthorized or illegitimate system user (an "outsider") from outside the perimeter. Potential outside attackers on the internet can range from inexperienced practical jokers to hostile governments, multinational terrorist organizations, and organized crime [15].

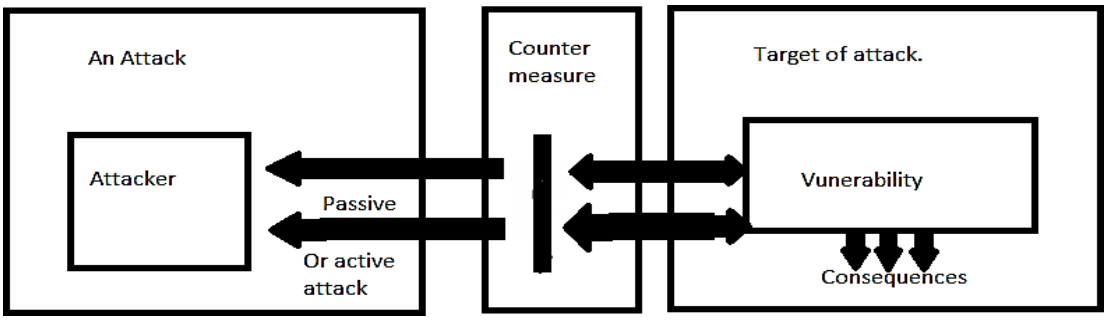

Figure 2.2: Type of Cyber Attack [16]

When an attack aims to modify system resources or interfere with their functionality, it canbe considered active and jeopardize availability or integrity. A "passive attack" jeopardizes confidentiality since it tries to obtain or utilize information from the system without affecting system resources.

A threat refers to the possibility of a security breach. It can be any situation, ability, action,or occurrence that has the potential to compromise security and result in harm. In simple terms, a threat is a potential harm that could exploit a weakness. A threat may be "accidental" (e.g., a

computer malfunction or the risk of a natural disaster like an earthquake, fire, or tornado) or "intentional" (e.g., perpetrated by an individual or a criminal organization) [15].

Physical attacks include things like computer theft and equipment damage. Others involveattempting to compel modifications to the logic employed by computers or network protocols to produce outcomes that were not intended by the original creator but were advantageous to the attacker. Malware is software that is used to attack computers logically.

Abinesh Kamal and Divya [16] describe the escalating threat landscape of cyberattacks andemphasize the critical need for organizations to establish robust security operations. The central focus is on integrating threat intelligence into security operations and leveraging machine learningtechniques for precise threat analysis. The authors propose an integrated threat intelligenceplatform, emphasizing precision and reliability in threat analysis. The success of the platform is evaluated through comprehensive performance metrics, highlighting its potential to enhancesecurity operations within organizations.

Beckerman [17] explores the discourse surrounding the potential risks of cyberattacks triggering a conventional war between states, focusing on the concept of a cyber-security dilemma.The author, Carly E Beckerman, aims to critically evaluate this dilemma, challenging existing notions and assessing the likelihood of unchecked escalation from state-led cyber competition to kinetic warfare. The study emphasizes the importance of understanding how cyber warfare may unexpectedly provoke escalation, not adequately addressed by current escalation models. Findingsand conclusions are derived from a thorough analysis of existing literature, theoretical frameworks,and case studies, offering insights into the cybersecurity dilemma.

Al-Hawawreh and Moustafa [18] address the escalating threat of cyber-physical attacks inIndustrial Internet of Things (IIoT) systems, emphasizing the potential for physical damage and human safety risks. Key issues include the lack of attack intelligence for physical processes, the complexity of cyber-physical attacks, and the need for real-time explanations. The authors proposean attack intelligence framework leveraging artificial intelligence, machine learning, deep learning,and Explainable AI (XAI) to identify and understand cyber-physical attacks, providing actionableinsights for security analysts.

Table 2.1: Literature Related to Cyber Attacks

Sr. no	**Author**	**Features**	**Methods/ Algorithm**	**Remarks**
01.	Kamal & Divya [16]	Robust security operations, precise threat analysis	Leveraging machinelearning, the UNSW-NB15 dataset	Enhancing security operations within organizations
02.	Beckerman [17]	Escalation of state-led cyber competition to kinetic warfare, focusing cyber-security dilemma.	Multi-level Neoclassical Realist(NCR) framework	Multi-faceted escalation cyber warfare, Adapts theEuropean Literacy Policy Network's (ELINET)
03.	Al-Hawawreh and Moustafa [18]	Cyber-physical attacks in IIoT	XAI-based module,LSTM, and DNN models	Identify indicators, distinguishing attack types.
04.	Wang et al. [19]	Detecting Return-Oriented Programming (ROP) attacks, trade-off between cost and security.	CNN-based ROPdetection model.	Reduction in false positive detection rate
05	Tongkacho k et al. [20]	The efficiency of traditional security solutions, detecting unpredictable cyber-attacks.	Leveraging machine learning,create autonomous and intelligent security systems	Bypass security protocols and social networks combined to anticipate potential assaults on specific organizations.

Wang et al. [19] address the scalability issue of deep learning-based approaches in cybersecurity, particularly in detecting Return-Oriented Programming (ROP) attacks. The key challenges involvethe impact of imbalanced datasets on model performance, the trade-off between cost and security,and biased models due to the large number of positive samples compared to negative samples. Theauthors propose using transfer learning to enhance the effectiveness, scalability, and practicality ofdeep learning approaches in real-world scenarios. The authors conducted evaluation experiments to test the proposed method. Results demonstrate the reduction in false positives through the proposed approach. Analyzing the trade-off on the detection rate, indicating the practical application of the method.

Tongkachok et al. [20] address the inefficiency of traditional security solutions in detectingnew and unpredictable cyber-attacks, especially with the rise of autonomous cyber-attacks. It aimsto explore the potential of machine learning in improving cyber security by enhancing the detectionand response to evolving threats. The authors explore the importance of quality information for effective machine learning in cybersecurity. Social media network characteristics and machine learning algorithms were combined to anticipate potential assaults on specific organizations.

2.4 Malware

Malicious software, also known as malware, refers to any software designed to interfere with computers, servers, clients, or computer networks. Its objectives include gaining unauthorizedaccess to systems or data, revealing private information, blocking access to data, or inadvertently compromising user security and privacy on their computer. Researchers often categorize malwareinto one or more subtypes. Examples of malware include computer viruses, worms, Trojan horses,ransomware, spyware, adware, rogue software, wipers, and key loggers [21].

Malware poses serious risks on the Internet for both individuals and organizations. According to the Symantec 2018 Internet Security Threat Report (ISTR), there were 669,947,865 types of malware in 2017, double the number in 2016. Malware attacks and other computer-relatedcrimes contribute to cybercrime, which is expected to grow by 15% annually and cost the world economy US$6 trillion by 2021. Since 2021, malware has been specifically designed to target critical infrastructure networks, such as the electricity distribution system [22].

Depending on the kind of malware, different defense strategies are used, but most can be lessened by putting firewalls and antivirus software into place, applying patches regularly, protecting networks from infiltration, creating frequent backups, and isolating compromised systems. Malware, however, can elude detection by antivirus software's algorithms.

2.4.1 Types of Malwares

There are numerous ways to categorize malware, some malicious software falls intosix overlapping categories.

Figure 2.3: Types of Malwares [23]

Virus

A computer virus is a kind of software that typically appears as an innocuous application inside another program. It can replicate itself and insert copies of itself into other files or programs,which frequently leads to destructive activities like erasing data. Because computer viruses can replicate and propagate on their own, they are frequently compared to biological viruses. A common example of this is a portable execution infection, which is a technique used to insert additional data or executable code into PE files, typically to spread malware [23]. In essence, a computer virus is a piece of software that enters another executable program like the operating system—into the target system without the user's knowledge or permission and then multiplies to other executable files when the program is executed.

Worm

A worm is a unique kind of malware that replicates itself without affecting files and runs on its own. Actively moving across networks, it infects other computers that come in its path. Thisdistinction suggests that a virus propagates through the user executing infected software or an infected operating system, whereas a worm spreads independently [24].

Rootkits

Malicious software must stay hidden after it is installed on a system to be discovered. Thisconcealment is made easier by rootkits, software programs that alter the host's operating system and conceal the virus from the user. Rootkits can hide harmful processes from being seen in the system's list of processes and restrict access to the files they contain. Certain malicious software implements procedures to avoid detection and removal attempts, going beyond simple concealment. The Jargon File story describes an early instance of this behavior, in which two appscompromised a Xerox time-sharing system. Every "ghost" task quickly learned when the other oneended and immediately started a fresh instance of the recently ended application, making it very difficult to get rid of both ghosts without purposefully breaking the system [25].

Backdoors

A backdoor is a type of computer software that grants an attacker unauthorized remote access to a victim's computer, often without the victim's awareness. Typically, attackers employ other methods, such as trojans, worms, or viruses, to circumvent authentication restrictions and install the backdoor application, commonly over an unprotected network like the Internet. Additionally, backdoors may arise from software vulnerabilities in reputable programs exploited by attackers to access a victim's computer system or network.

It has been suggested frequently that computer manufacturers may pre-install backdoors ontheir systems to facilitate technical support for customers. According to a 2014 report, US government agencies were purportedly sending computers purchased by individuals identified as "targets" to covert workshops, where they installed hardware or software enabling remote access to the computers. This was reportedly among the most effective methods used by these agencies togain access to global networks. Backdoors can be installed using various techniques, including trojan horses, worms, implants, and others [26].

Trojan horse

A Trojan horse disguises itself as a legitimate application or utility to deceive victims into installing it. Usually, it harbors a hidden damaging feature that activates when the application is launched. The term derives from the Ancient Greek myth of the Trojan horse, which was employedto infiltrate Troy and launch an attack. Trojan horses are commonly distributed through methods like drive-by downloads or social engineering tactics, such as tricking users into opening seeminglyharmless email attachments, like a standard form requiring completion.

While the payload of a Trojan can vary widely, many recent versions operate as a backdoorby establishing communication with a controller, allowing unauthorized access to the compromisedsystem. Additionally, they may install additional software, such as adware or cryptocurrency mining software, to generate revenue for the Trojan's owner. However, detecting Trojan horses andbackdoors can be challenging on their own. When cryptocurrency mining software is installed, computers may exhibit symptoms such as slower performance, increased heat generation, or louderfan noise. To avoid detection, cryptocurrency miners may limit the resources they consume or operate only when the system is inactive. Unlike computer viruses and worms, Trojan horses typically do not attempt to insert themselves into other files or spread through other means [27].

Droppers

Trojan subtypes known as droppers are specifically designed to infect a system and introduce malware into it. They aim to avoid detection by being stealthy and keeping the payload small. It's important to distinguish between a dropper and a loader or stager. A loader or stager simply loads an extension of the malware into memory, such as a set of malicious functions throughreflective dynamic link library injection. The objective is to keep the initial phase subtle and lightweight. In contrast, a dropper's sole function is to infect the system with additional malware [28].

Ransomware

Ransomware is a type of malicious software that blocks access to computer files or the entire system until a ransom is paid. There are two main types of ransomwares: crypto-ransomware and locker ransomware. Crypto-ransomware not only locks down a computer system but also encrypts its contents, while locker ransomware only restricts access to the

system. For example, applications like Crypto Locker encrypt files securely, and decryption is only possible upon payment of a significant sum [29].

Malware can be used to commit click fraud by simulating a computer user clicking on an advertisement link on a website and receiving payment from the advertiser. An estimated 60-70%of malware still in circulation is used for click fraud, and approximately 22% of all ad clicks are false [30].

Screen lockers, sometimes known as lock screens, are a kind of "cyber police" ransomwarethat purports to be gathering illicit content on Windows or Android devices and blocks the screensin an attempt to coerce victims into paying a ransom. More Android devices are affected by Just and SLocker than by other lock screens; Just accounts for about 60% of all ransomware detectionson Android devices. As the name implies, encryption-based ransomware is a kind of malware thatencrypts every file on a compromised system. The user is then informed by a pop-up window by these malware types that their data have been encrypted and that they need to pay (typically in Bitcoin) to get them back. Crypto Locker and WannaCry are two instances of malware that use encryption [30].

Varadharajan et al. [31] used a Genetic Algorithm (GA) as the technique to optimize Convolutional Neural Networks (CNNs) for image-based malware classification. The criteria used in the paper include optimizing the architecture and hyper-parameters of the CNN classifier, evaluating the performance of the evolved networks on benchmark malware datasets, and comparing the performance of the GA-designed networks to state-of-the-art results in image-basedmalware classification.

Agarap [32] describes in the paper "Towards Building an Intelligent Anti-Malware System: A Deep Learning Approach using Support Vector Machine (SVM) for Malware Classification" using deep learning techniques include the evaluation of different deep learning models such as CNN-SVM, GRU-SVM, and MLP-SVM in a multinomial classification task using the Malimg dataset.

The performance of these models is assessed based on their predictive accuracy on the test dataset, with the GRU-SVM model achieving the highest accuracy. Additionally, the potential improvements to the architecture design of the CNN-SVM and MLP-SVM models by adding more hidden layers, better nonlinearities, and optimized dropout to enhance their performance inmalware classification.

Zhong et al. [33] concluded in the paper "Malware-on-the-Brain: Illuminating Malware Byte Codes with Images for Malware Classification," where the authors employ a novel techniquethat converts malware samples into images for classification. This technique involves utilizing contrast-limited adaptive histogram equalization (CLAHE) to enhance the visual representation ofmalware byte codes. Furthermore, the authors evaluate the performance of their VisMal framework to improve the discernibility between different malware families and facilitate the classification process.

Table 2.2: Literature Related to Malware Attacks

Sr. no	**Author Name**	**Features**	**Methods/ algorithm**	**Remarks**
01.	Varadharajan etal. [31]	Architecture Optimization Hyper-parameter Optimization	GA's	GA optimization techniques are inspiredby the process of natural selection. GAsoperates on a population of potential solutions.
02.	Agarap [32]	Deep Learning Models with SVM	CNN-SVM GRU-SVM MLP-SVM	The CNN is used forfeature extraction.GRUs, a recurrentneural network (RNN),are used to handle sequential data. The MLP is used to learn representationsfrom the input data.
03.	Zhong et al [33]	VisMal Framework	Image Enhancement using CLAHE	The framework is designed to improvethe discernibility between different malware families byenhancing the visual features of the malware images.

2.5 Tor Browser

The Onion Router, also known as Tor, is a tool that enables anonymous communication by routing Internet traffic through a global network of over 7,000 volunteer relays, all free of charge.It is more difficult to track down an individual's online activities when they utilize Tor. By hidinga user's location and usage from anyone doing network monitoring or traffic analysis, Tor preserves individual privacy. Leveraging Tor exit nodes safeguards the user's freedom and capacity for private communication by leveraging IP address anonymity.

Many individuals use Tor for both lawful and unlawful activities, as it provides the ability for users to anonymously browse the Internet, chat, and send instant messages. For example, law enforcement organizations, hacktivism groups, and criminal enterprises have all used Tor against each other. Tor is utilized for illicit purposes as well. These can include the dissemination of contentthat depicts child abuse, the selling of drugs, the protection of privacy, and the evasion of censorship[34].

Eaton et al. [35] address privacy challenges faced by Tor onion services, focusing on the vulnerability of onion services to privacy-compromising attacks by potentially malicious Hidden Service Directories (HSDirs). The primary concern is the risk of compromising the privacy and unlikability of Onion services, making them susceptible to tracking over time. The proposed solution involves integrating Private Information Retrieval (PIR) into the descriptor lookup processto conceal information about queried services, even from untrustworthy HSDirs. The ultimate goalis to enhance the privacy and anonymity of Onion service users.

Bergman and Oliver [36] address the challenges posed by the use of anonymous communication networks, particularly the Tor network, by cybercriminals engaging in illegal activities within the Dark Web. The key issues include the difficulty faced by law enforcement in identifying and combating cybercrimes due to the anonymity provided by these networks and the lack of human resources. The paper specifically focuses on recognizing Tor-related malware and onion services within the Dark Web. The achieved high accuracy rates in identifying Tor-related malware and onion services demonstrate the effectiveness of the proposed approach in mitigating challenges associated with the anonymity of the Tor network and criminal activities within the Dark Web. Machine learning classifiers, including decision tree classifiers and logistic regressionalgorithms, achieved over 90% accuracy in identifying and classifying Tor-related malware and onion services.

Huang and Yanhui [37] describe the vulnerability of Tor onion services to circuit fingerprinting attacks. These attacks aim to classify circuits into those generated by the client and those generated by the Onion service, ultimately revealing network addresses and threatening the anonymity provided by Tor. The authors propose techniques SVM, Random Forest, and XGBoostto predict the identity of the host and classify circuits by type. The results obtained from these experiments form the foundation for the proposed novel circuit fingerprinting attack and the evaluation of the performance of machine learning classifiers in identifying and classifyingdifferent circuit types.

Ödén and Björnberg [38] aim to investigate and propose a method to deanonymize criminals using digital watermarks in the Tor network. The goal is to track illegal activities on theTor platform. The paper discusses the use of different data collection tools, analyzes results, and explores potential implications, risks, and mitigations associated with the proposed deanonymization method.

Wang et al. [39] aim to provide a thorough and long-term evaluation of Tor V3 onion services. The primary focus is on accurately estimating the size, analyzing the ecosystem, and addressing challenges related to evaluating the popularity of V3 onion services. The assessment covers a range of onion services, including benign services, abuses, and black markets. The authorstake into account factors such as online rate, access frequency, and the presence of onion addressesin the Yellow Pages. The results include both emulated and real-world validations, enhancing the credibility and applicability of the proposed estimation methods and algorithms. Deploying HSDirson the actual Tor network to observe and validate estimation methods in a live environment.

Table 2.3: Literature Related to Tor Network

Sr. no	Author Name	Features	Methods/ Algorithm	Remarks
01.	Eaton et al. [35]	Vulnerability of onionservices, potentially malicious HSDirs.	Newer version 3 onionservices, PIR-based solution.	Micro benchmarks live Tor network, Indicated real- world testing.
02.	Bergman andOliver [36]	Cybercriminals engaging in illegal activities within the Dark Web	Tor-related malware, onion services.	Achieved accuracy in logistic regression algorithms
03.	Huang and Yanhui [37]	Vulnerability of Toronion services to circuit fingerprinting attacks	SVM, Random Forest,and XGBoost	Identify novelcircuit fingerprintingattack.
05.	Ödén and Björnberg [38]	Deanonymize criminals using digitalwatermarks in the Tornetwork.	Axel's thesis (ACKpackets)	Exploit weaknesses in deanonymization techniques, discover watermarks in thepackets, and trackillegal activities.
06.	Wang et al. [39]	Long-term evaluationof the popularity of Tor V3 onion services	DOM trees	Enhance credibility androbustness**,** Emulated andreal-world environments estimated.

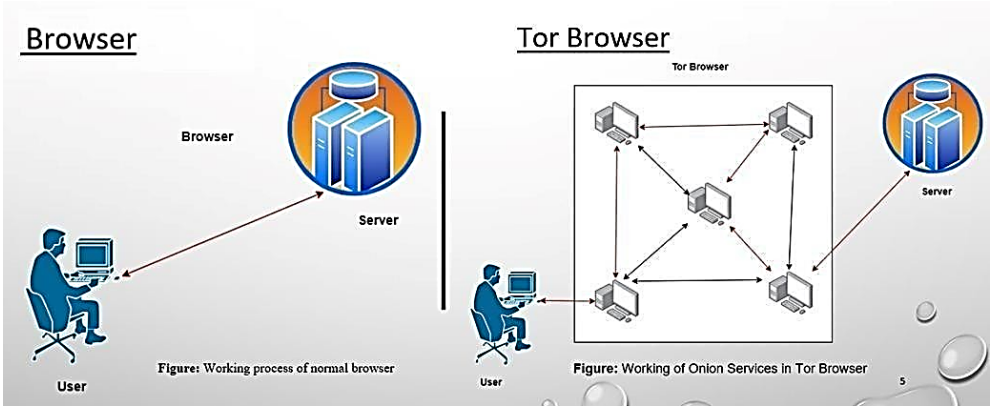

Figure 2.4: Difference Between Normal Browse And Tor Browser

2.6 Onion Services

Tor can provide anonymity for not only users but also for servers and web pages. Onion services, previously known as hidden services, are servers that are set up to only accept connections through Tor. These services can be accessed using an onion address, usually through the Tor Browser, instead of revealing the server's IP address and network location. The Tor network identifies these addresses by finding matching public keys and introduction points stored in a distributed hash table across the network. This system ensures the anonymity of both parties whilehandling the routing of data to and from onion services, even those behind firewalls or network address translators (NAT). Accessing these onion services requires the use of Tor [40].

Onion services, initially outlined in 2003, have been part of the Tor network since 2004. Tor operates on a decentralized model, except for the database containing onion service descriptors.While various onion services list publicly available onion addresses, there isn't a comprehensive, easily accessible directory of all onion services. To facilitate the discovery of content hosted on websites within the Tor network, an online search engine called Tor Search indexes pages [41].

Despite the end-to-end encryption of connections to onion services, which renders them impervious to eavesdropping due to routing through the Tor network, security issues persist with Tor onion services. For instance, services accessible through both Tor onion services and the publicInternet are susceptible to correlation attacks, thus lacking complete concealment.

2.7 Cyber Threat Intelligence

Cyber Threat Intelligence (CTI) aims to proactively prevent potential cyber-attacks and adverse events by offering insights, expertise, and information based on knowledge and experience. This encompasses assessing both physical and cyber threats, as well as understanding threat actors. Sources of cyber threat intelligence encompass open-source intelligence, social media intelligence,human intelligence, technical intelligence, device log files, forensically obtained data, internet traffic information, and data collected from the deep and dark web.

Threat intelligence has grown in importance in recent years as a component of commercialcyber security strategies because it enables organizations to take a more proactive stance and identify which threats pose the biggest hazards to their operations. As a result, businesses adopt a more proactive stance, deliberately seeking out their weaknesses and thwarting hackers before theyoccur. This technique is becoming more and more popular since, according to IBM estimates, 47%of all assaults against corporations involve threat exploitation [42].

The COVID-19 pandemic and an increase in remote work have also contributed to an increase in threat vulnerabilities in recent years, which increases the vulnerability of company data.Many businesses have chosen in recent years to outsource their threat intelligence efforts to a managed security service (MSSP) as a result of the increasing sophistication required for threat intelligence, on the one hand, and the escalating threats on the other.

The intelligence cycle, which consists of five phases, is a continuous and cyclical process that intelligence teams use to create cyber threat intelligence. The goal of the cycle is to give leadership useful and pertinent intelligence that minimizes risk and uncertainty [43].

2.7.1 Classes of Cyber Threat Intelligence

There are three main classes of cyber threat intelligence [44].

Tactical Usually used to assist in locating potential threats. Cybercriminals employ indicators of compromise, such as IP addresses, Internet domains, or hashes, and a deeper examination of their strategies, methods, and procedures (TTP) is starting to take place. Security teams will be able to anticipate assaults and detect them early on thanks to insights produced at

thetactical level [44].

Operational The highest level of threat intelligence is this one. It provides precise information on assaults, the driving forces behind them, the capabilities of threat actors, and particular campaigns. At this level, threat intelligence specialists offer insights into the characteristics, intentions, and timing of new threats. This kind of information is harder to come byand is typically gathered via hidden, inaccessible online forums that are inaccessible to internal teams. Teams responsible for security and attack response employ this kind of operational intelligence [44].

Strategic Information about the overall risks associated with cyber threats is usually gearedtoward non-technical audiences. The intention is to provide executives with a prioritized list of actions by providing a thorough study of both present and anticipated future business risks in the form of white papers and studies, along with the possible repercussions of threats [44].

Kasowaki and Kuzey [45] revolve around addressing the escalating complexity and diversity of cyber threats in today's digital landscape. The authors emphasize the critical need fora proactive cybersecurity approach, specifically through the understanding, identification, and mitigation of cyber risks using effective threat intelligence practices. The paper involves a comprehensive exploration of the essence and application of threat intelligence in cybersecurity. The authors used techniques to provide a detailed overview of threat intelligence, cyber risks, andcybersecurity frameworks. It evaluates how well threat intelligence can address emerging threats, and vulnerabilities, contribute to cybersecurity defenses, and assesses the overall impact of threat intelligence practices on cybersecurity resilience and incident response capabilities.

Adeniyi and Ness [46] discuss the growing complexity and prevalence of cyber threats in the digital realm. The authors stress the importance of implementing advanced cybersecurity measures and suggest incorporating artificial intelligence (AI) as a key tool to improve threat detection, automate response mechanisms, and strengthen organizations against evolving cyber threats. The authors used an investigative approach to explore the uses, benefits, challenges, and evolving landscape of artificial intelligence in cybersecurity.

Yilmaz and Kasowaki [47] address the escalating challenges of cyber threats, emphasizingthe crucial role of Cyber Threat Intelligence Analysts in mitigating these risks. The

paper focuses on understanding the complexities of cyber threats, the necessity for skilled professionals, and themethodologies analysts use to protect digital infrastructures.

It explores the critical functions of Cyber Threat Intelligence Analysts, including data gathering and analysis to identify patterns, trends, and indicators of compromise (IoCs) for assessing potential threats. The authors highlight collaborations with cybersecurity teams to develop proactive defense strategies and actionable recommendations. Advanced analysis techniques are employed to identify patterns and assess the severity of threats, prioritizing responses and countermeasures. The paper is primarily theoretical and descriptive, concentrating on the roles, methodologies, and responsibilities of Cyber Threat Intelligence Analysts rather thanpresenting experimental results.

Rahman et al. [48] address the challenge of predicting temporal relations between cyber threat techniques. The central problem is formulated as a text classification task, where machine learning models are used to predict ATT&CK techniques based on cyber threat intelligence reports.The goal is to overcome challenges related to identifying multiple techniques in a single sentence,sentences that may not describe any technique, and the need for a sufficient number of examples for training a text classifier for the 193 techniques listed in ATT&CK. The focus is on advancing the understanding of temporal attack patterns through systematic analysis and leveraging machinelearning models for improved prediction capabilities in the cybersecurity domain.

Sakellariou et al. [49] study the difficulties that current cyber threat intelligence (CTI) systems encounter in effectively using discussion forums as a source of raw data. They present theSECDFAN (SECurity Discussion Forums Analysis) system, which aims to standardize the creationand distribution of CTI products from raw data gathered from discussion forums on the surface web, deepnet, and darknet. The main focus is on using a systematic design approach, creating semantic schema, and conducting content analysis to address the limitations of existing CTI systems.

Sanger et al. [50] concentrate on the development of a proactive cyber threat intelligence system geared towards detecting cybercrimes within Dark Web forums. The paper highlights the application of techniques such as NLP and deep learning, the implementation of a carefully crafted annotation scheme, and the attainment of high accuracy in classification models. Notably, their BERT-based classification model achieves an impressive accuracy score of 96%,

underscoring theeffectiveness of their tailored preprocessing strategies and the validation of their annotation scheme.

Ahmed et al. [51] explore the challenges linked to obtaining useful insights from Cyber Threat Intelligence (CTI) data. They aim to provide security professionals with more accurate andtimely insights to support informed decision-making processes. They evaluated the model's performance using metrics like precision, recall, and F1 score. The results showcased the effectiveness of the joint extraction technique, with comparisons against existing models highlighting its superiority in terms of accuracy, scalability, and generalizability. These findings make significant strides in overcoming challenges associated with CTI data extraction, empowering security professionals with more dependable insights for decision-making.

Dekker and Alevizos [52] address the challenge of decision-making under uncertainty in information security, particularly in the context of the continuously evolving cyber threat landscape. The authors propose a threat-intelligence-driven methodology called Threat-Intelligence Based Security Assessment (TIBSA) to address uncertainty in cyber risk analysis and enhance decision-making in information security. Key components of the TIBSA methodology include utilizing cyber threat intelligence (CTI) to inform the assessment process at strategic, tactical, and operational levels. The paper focuses on improving cybersecurity posture and decision-making capabilities by leveraging threat intelligence and analytical techniques to addressuncertainties in the cyber threat landscape.

Sun et al. [53] delve into the increasing severity and frequency of cyber-attacks, underliningthe necessity for innovative security measures in response to evolving threats. Recognizing the inadequacy of traditional security approaches, the authors advocate for the exploration of Cyber Threat Intelligence (CTI) mining as a proactive defense strategy. This framework facilitates the transformation of raw data into actionable intelligence, empowering organizations to make well- informed decisions and fortify their cybersecurity defense strategies. The paper emphasizesanalyzing and categorizing existing research endeavors about CTI mining techniques, knowledge acquisition, and cybersecurity defense strategies.

Samtani et al. [54] address the need for Explainable Artificial Intelligence (XAI) in the field of Cyber Threat Intelligence (CTI). The main issue is the lack of transparency, interpretability,andtrustworthiness in AI-based CTI systems. The paper focuses on developing techniques and modelsthat provide explanations for AI system decisions in CTI. This enables

analysts and stakeholders to understand and validate the reasoning behind the generated intelligence.

Moraliyage et al. [55] address the challenges presented by the increasing number of onion services on the dark web, including the anonymity and seriousness of cybersecurity threats arising from illegal activities and the significant rise in illicit onion services. They emphasize the importance of monitoring the dark web to proactively identify potential cybersecurity threats through Cyber Threat Intelligence (CTI). Their method uses a Convolutional Neural Network (CNN) with Gradient-weighted Class Activation Mapping (Grad-CAM) for image classification and pre-trained word embedding with Bahdanau additive attention for text classification. With a modular architecture, this approach can be easily incorporated into threat-sharing platforms, improving the sharing of threat intelligence across different platforms. Additionally, the authors utilize explainable deep learning techniques such as Grad-CAM and attention visualization to provide insights into the decision-making processes.

Disha and Waheed [56] aim to evaluate the performance of machine learning models for intrusion detection systems (IDS) using contemporary datasets representing modern cyber threats. The primary challenge addressed is the effective detection and prevention of cybercriminal activities. The study compares traditional machine learning techniques with modern approaches, tocontribute to cybersecurity by assessing the effectiveness of these models and providing insightsintotheir strengths and limitations for intrusion detection. The proposed methodology includes the selection of machine learning models and the use of the Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique, which is crucial for addressing feature selection challenges. The findings contribute to advancing intrusion detection systems and cybersecurity measures in the context of modern cyber threats.

Alsaedi et al. [57] address the challenge of developing a robust malicious URL detection model using cyber threat intelligence (CTI) and ensemble learning techniques. It aims to enhance detection performance and reduce false-positive rates compared to existing URL-based models. The high dimensionality of features from Google data is acknowledged, leading to the need for selecting features representing URLs effectively using information gain. Highlights the dynamic nature of URL-based features, subject to manipulation by attackers, making them insufficient for effective representation.

Xin Ge et al. [58] have developed a systematic approach for identifying security risks in embedded devices by focusing on firmware fingerprinting. They aim to remotely detect firmware versions, extract vulnerabilities, and assess security risks. They created a prototype system and conducted real-world experiments to validate their approach. They used criteria such as similaritythresholds, precision, recall, and threshold values to optimize and evaluate performance. The firmware fingerprinting achieved 91% precision and 90% recall. The real-world experiments confirmed the accuracy and effectiveness of the proposed methodology, providing valuableinsights into firmware vulnerabilities and associated risks.

Herron and Hally [59] address the security challenges arising from the integration of IoT devices and wireless technologies into business environments. The primary focus is on providing recommendations and implementation techniques to enhance security controls and mitigate vulnerabilities associated with IoT wireless technologies. The paper contributes recommendationsand practical techniques to address security challenges stemming from IoT wireless technologies in business environments.

Omar Batarfi et al. [60] address the increasing threat of ransomware attacks, focusing on the need for effective cyber threat-hunting techniques to detect and respond to these evolving cyber threats. The primary objectives include providing a systematic review of existing literature,techniques, and cyber threat intelligence (CTI) approaches related to ransomware detection,identifying limitations, and proposing new directions for research and development. The paper contributes a systematic review of cyber threat hunting techniques for ransomware detection, addressing the evolving nature of ransomware attacks.

Yeboah-Ofori et al. [61] address the challenge of predicting cyber-attacks in the context ofCyber Supply Chain (CSC) organizations. The authors emphasize the necessity for these organizations to utilize Machine Learning (ML) analytics coupled with Cyber Threat Intelligence (CTI) to predict, analyze, and control cyber threats effectively. The paper provides a conceptual view of CSC system security and proposes a process for threat analysis, prediction, and control. Through experiments and validation, the authors provide insights into the effectiveness of their approach and highlight the most predictable threats in the CSC domain. The focus is on enhancingcybersecurity by combining intelligence-driven threat analysis with machine learning capabilities.

Moustafa et al. [62] focus on the study and analysis of web application attacks. The

primarygoal is to develop a methodology for capturing web attack data, extracting relevant features, modeling the attack data, and simulating it. The overarching aim is to improve cybersecurity by understanding and predicting web attacks. The focus is on capturing, extracting, modeling, and simulating web attack data to enhance threat intelligence and improve cybersecurity.

Holler et al [63] address potential privacy issues associated with long-lived onion service addresses in decentralized systems. The problem statement emphasizes the privacy concerns related to persistent onion service addresses, which can be exploited as identifying metadata. To tackle this issue, the paper explores the feasibility of using dynamic, short-lived onion services asa solution to minimize metadata leakage and enhance privacy in decentralized systems. The research aims to provide insights into the performance impact and practicality of constantly generating new onion services as a privacy-enhancing measure. The combination of simulations and physical testing contributes to a comprehensive assessment of the proposed solution. Feasibility of dynamic onion services assessed based on provisioning times.

Tudu et al. [64] address the cyber-security risks associated with deploying open-source digital library systems, focusing on DSpace and Greenstone. The objective is to identify security deficiencies in these widely used systems and provide recommendations for improvements. The authors aim to categorize vulnerabilities, assess severity levels, and recommend changes to enhance the overall security of DSpace and Greenstone. Through port scanning, static and dynamic analysis, and code/configuration reviews, the authors identify vulnerabilities and provide recommendations.This emphasis is on enhancing the security of these widely used open-source digital library systems.

Albahar et al [65] aim to tackle the growing threat of cyber-attacks by developing and comparing various neural network-based models for detecting and classifying malicious activities in cyber-physical systems. The authors emphasize the need for efficient intrusion detection systemsto mitigate the impact of cyber threats and highlight the importance of neural network methods in cybersecurity applications. The focus is on evaluating the performance of different neural network models using a cyber-physical subsystem dataset. The results are obtained through experiments onreal-life data, emphasizing the effectiveness of specific neural network models for intrusion detection in cybersecurity applications.

Zarandi and Sharif [66] focus on addressing the challenge of detecting and mitigating

cyber-attacks in distributed control systems, emphasizing the identification and isolation of misbehavingagents in interconnected networks. The primary goal is to develop a resilient distributed control algorithm capable of real-time attack detection and response to ensure system stability and security. The proposed solution involves the use of a reputation manager, a consensus manager, and a deepneural network for monitoring agent behavior, making decisions on reputation and connectivity, and analyzing data for attack detection. Evaluation criteria include accuracy, mean squares of error, and overall effectiveness in addressing the cybersecurity challenges in distributed control systems.

Table 2.4: Literature Related to Cyber Threat Intelligence

Sr. no	**Author Name**	**Features**	**Methods/ Algorithm**	**Remarks**
01.	Kasowaki &Kuzey [45]	Escalating complexity and diversity of cyberthreats in today's digital landscape	Simulations andphysical testing	Emphasize a descriptive and analytical approach
02.	Adeniyi & Ness[46]	Measures and integration of AI	Conducts a literature reviewand discusses ethical considerations	Focus on conceptual exploration and analysis of the role ofAI in cybersecurity.
03.	Yilmaz and Kasowaki [47]	Explorations and complexities of cyber threats	IoCs to assess potential cyberthreats.	Emphasizes the significance and collaboration of proactive threat analysis,

04.	Rahman et al.[48]	Predicting temporal relations between cyber threat techniques	Systematic analysis and leveraging machine learning models	Findings derived from analysis of cyber threat intelligence reports
05.	Sakellariou etal. [49]	Effectively utilizing forums as a sourceof raw data	SECDFAN system	SECDFAN physical testing to obtainspecific results
06.	Sanger et al.[50]	Identifying cybercrimes within Dark Web forums	Agora Dark Webdataset, heuristic preprocessing novel multiclass	Tailored preprocessing strategies, in-depth discussion
07.	Ahmed et al.[51]	Extracting actionableCTI data.	Novel, classical pipeline techniques.	Security professionalswith more reliable insights for decision-making.
08.	Dekker and Alevizos [52]	Uncertainty in information security	Traditional risk analysis TIBSA	Enhance decision- making and tacticaloperational levels.
09.	Sun et al. [53]	Escalating severity and frequency of cyber-attacks	Cyber scenario analysis, CTI knowledge acquisition	Current state-of-the- art solutions, proactive cybersecurity defense.

10.	Samtani et al.[54]	Need for XAI, lack of trustworthiness in AI-based CTI systems.	LIME and Saliency maps.	Discovered leakage points in side-channelattack data
11.	Moraliyage etal. [55]	Anonymity and severity threats originating from illegal activities on the dark web	CNN and Grad-CAM	Provide insights into decision-making processes, and detecting cybersecurity threats.
12.	Disha and Waheed [56]	Intrusion detection systems (IDS), contemporary datasets	Adaboost, GBT, MLP, LSTM andGRU	Improving cybersecurity measures enhancesthe ability to detectand prevent cyber threats.
13.	Alsaedi et al.[57]	Robust malicious URL detection model	BFGS algorithm,ANN prediction model	Enhance performanceand reduce false- positive rates.
14.	Xin Ge et al.[58]	Vulnerabilities embedded devices through fine-grained firmware fingerprinting.	Web Crawling	Real-world experiments validate that firmware fingerprints are achieved.
15.	Herron and Hally [59]	Lack of device-levelcontrols, insufficient security controls in IoT protocols	Mapping Techniques CIS-CSC.	Reduction of attacksurface in business network vulnerabilities recognized securitycontrol frameworks.

16.	Omar Batarfi et al. [60]	Increasing threat ransomware attacks, significant harm to mitigation efforts	Ransomware, CTH techniques	Detecting both knownand unknown ransomware
17.	Yeboah-Ofori etal. [61]	Predicting cyber- attacks in the contextof CSC organizations	ML techniques predict CSC systems.	Spyware/Ransomwareand spear phishing were predicted in the CSC domain.
18.	Moustafa et al.[62]	Analysis of web application attacks, capturing web attackdata	Sophisticated OGM technique	Original data is betterthan other techniques,achieving a high DR and low FAR.
19.	Holler et al.[63]	Potential privacy issues associated with long-lived onion service, exploited as identifying metadata.	Peer-to-peer communication, the Tor network minimizes metadata leakage.	Feasibility of dynamiconion services assessed provisioningtimes.
20.	Tudu et al. [64]	Deploying open- source digital library systems, focusing on DSpace and Greenstone	Nmap, Nessus, Metasploit, SonarQube, Zed Proxy Attack (ZAP)	Enhanced security of DSpace and Greenstone, widely used open-source digital library systems
21.	Albahar et al.[65]	Tackle the growing threat of cyber-attacks, and distinguish differentneural network- based models.	Neural network-based models GRNN, PNN, RBNN, FFNN, ENN	Analyzed the effectiveness ofneural networkmodels

22.	Zarandi andSharif [66]	Detecting misbehaving agentsin interconnected networks	Deep neural networks, resilientcontrol algorithms	Evaluated the network's correctness and performance

2.8 Summary

This chapter offers a comprehensive review of the research background relevant to the study, encompassing existing literature on the internet, internet layers, internet security, cyber-attacks, types of malwares, onion services, and the Tor browser. It also examines related work andtechnologies focused on detecting cyber-attacks.

CHAPTER 3

RESEARCH METHODOLOGY

This chapter explains the research approach, detailing the study's methodology and the rationale behind the chosen techniques. It describes the methods used for the experiment, includingfeature extraction, feature selection, and data pre-processing. Additionally, it outlines the source ofthe experiment's data, as well as the implementation and evaluation process of the model.

3.1 Context

This research delves into the utilization of Convolutional Neural Networks (CNNs) to classify malware images into distinct categories. The CNN architecture comprises five convolutional blocks, each characterized by a defined number of convolutional layers and accompanying max-pooling layers. Regularization techniques, specifically L2 regularization, havebeen incorporated to combat overfitting challenges inherent in such models. The model underwenttraining on a substantial dataset, and its performance was evaluated using a suite of custom metricstailored for malware classification. These metrics encompass precision, recall, accuracy, specificity, conditional average metric, sensitivity, and F1 score. Throughout the training process, various callback mechanisms were implemented for monitoring and optimization. This research strives to contribute to the advancement of malware image classification techniques and underscores the significance of regularization and comprehensive evaluation metrics in achieving robust and reliable model performance.

3.2 Experiment Setup

A corei5 12th generation system with 16GB of RAM, 512 GB of SSD storage, and a 4 GBNVidia GPU is utilized for simulator purposes. Anaconda and Jupiter Notebook running Python are the tools employed for model development. Various libraries are used during the simulation process including TensorFlow, Keras, Pandas, Matplotlib, etc. The dataset was loaded and split into different ratios for testing the model's effectiveness in different scenarios.

3.3 Scope of Experiment

The main goal of the experiment is to create a new algorithm that can enhance the detectionand analysis of cyber threats within Onion services. This may include using machine learning techniques, statistical analysis, data mining, or other computational methods to process large amounts of data and pinpoint patterns that indicate malicious behavior. The goal of the experimentis to enhance the capability of cyber threat intelligence systems to detect and respond to threats targeting Onion services more effectively. The process may involve creating algorithms that can

identify new threats, connecting different data sources to find hidden threats, or improving the accuracy of threat classification and prioritization. Once the algorithms are developed, they need to be thoroughly tested, evaluated, and validated to ensure that they are effective and reliable in real-world situations. This might include using simulated environments and real-world data sets orworking with cybersecurity experts to assess how well the algorithms perform.

3.4 Datasets

The dataset comprises 125,001 PNG grayscale images from 18 malware families and benignsoftware some are represented in Fig 3.1. This research aims to build an efficient image classification model for distinguishing between different types of malware.

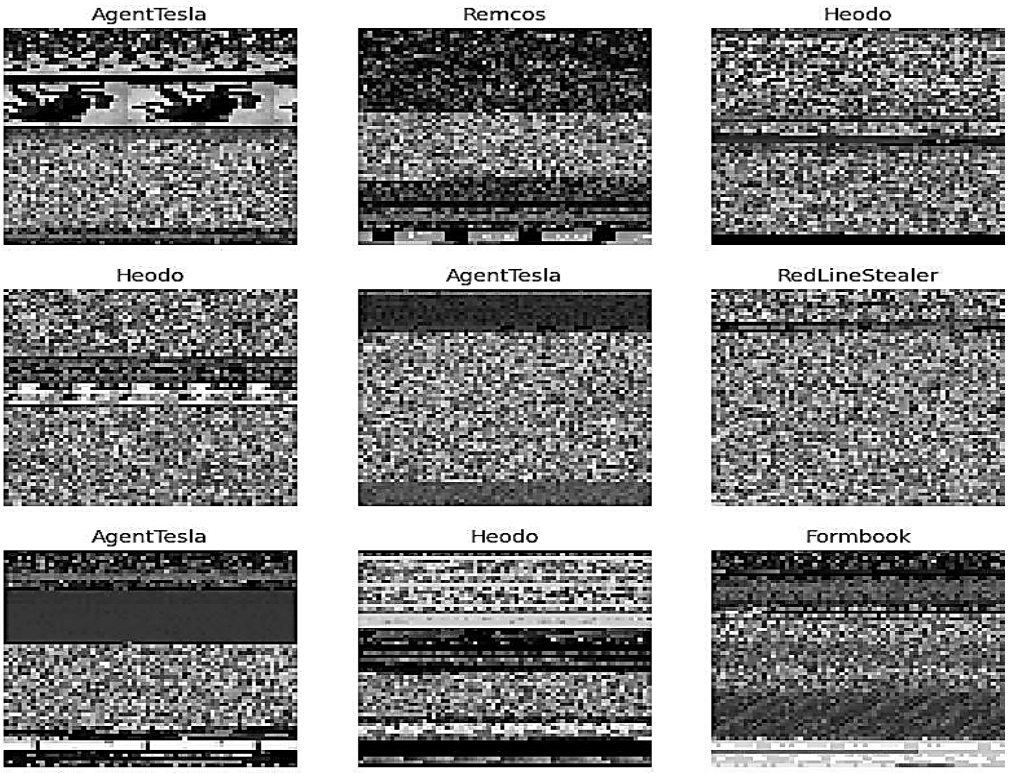

Figure 3.1: Gray Scale Image Sample

Figure 3.2 includes 18 classes in the dataset, consist of AgentTesla, Benign, CoinMinerXMRig, Danabot, Dridex, Formbook, Gh0stRAT, Glupteba, Gozi,Heodo, NanoCore, Quakbot, RecordBreaker, RedLineStealer, Remcos, Tinba, Trickbot, and Zeus[67].

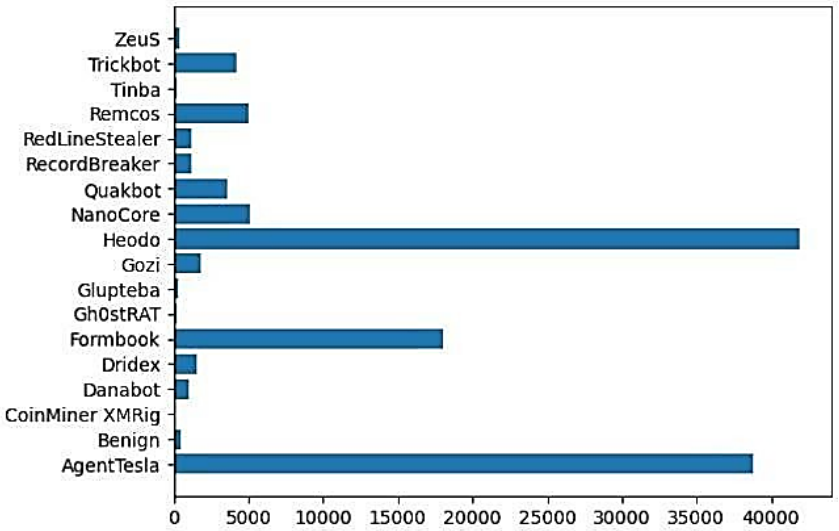

Figure 3.2: Classes View Data Description

3.4.1 Distributions of Datasets

The dataset contains a large amount of data, totaling 63 GB, split into 100,001 training filesand 25,000 validation files. Data visualization techniques were used to gain insights into the classdistribution within the dataset [64].

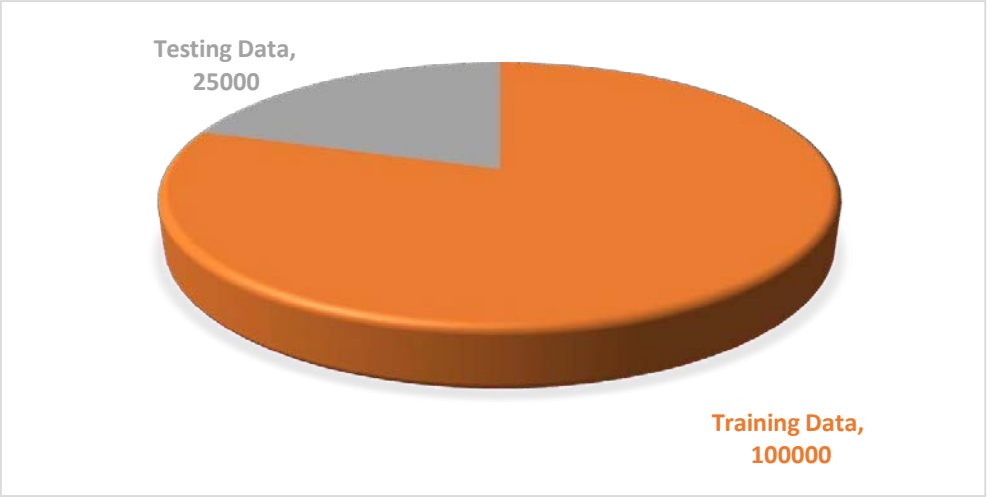

Figure 3.3: Splits Percentage of Dataset

3.5 Model Architecture

The focus of this research is the novel use of Convolutional Neural Networks (CNNs) to tackle the complex task of categorizing malware images into different groups. The selected CNN model is carefully crafted, with five convolutional blocks, each equipped with a distinct arrangement of convolutional layers and accompanying max-pooling layers. This section providesin-depth details about the architecture, highlighting the reasoning behind the design decisions madeto enhance the model's performance.

3.5.1 Convolutional Block Design

The model architecture starts with several convolutional blocks. The first four blocks havetwo convolutional layers each, while the fifth block has three convolutional layers. This arrangementis designed to help the model capture different features from the input images, including both basicand advanced characteristics needed for precise classification.

3.5.2 Addressing Overfitting with L2 Regularization (0.001)

To address the issue of overfitting in complex models, L2 regularization is introduced as akey component. This technique imposes constraints on the model weights, reducing the risk of fitting noise in the training data. Incorporating L2 regularization is a deliberate step to improve thegeneralization capabilities of the CNN model.

3.5.3 Max-Pooling Layers

In each convolutional block, there are strategically placed max- pooling layers. These layersare important for decreasing the size of the feature maps, simplifying the computations, and improving the model's ability to recognize patterns regardless of their position. The selection of max-pooling layers is crucial for maintaining important features while reducing the influence of unimportant differences.

3.5.4 Flattening and Dense Layers

After the convolutional blocks, a flattened layer is used to convert the 3D feature maps intoa one-dimensional array. Then, two dense layers, each containing 4096 neurons, enhance the model's ability to learn complex patterns and relationships within the data. This multi-layered architecture enables the extraction of complex feature hierarchies.

3.5.5 Output Layer and Dropout (0.4)

The model is finalized with an output layer comprising 18 nodes, representing the distinct malware classes. To prevent overfitting, a dropout layer is strategically inserted, randomly droppingout neurons during training, forcing the model to rely on a diverse set of features and thereby improving robustness.

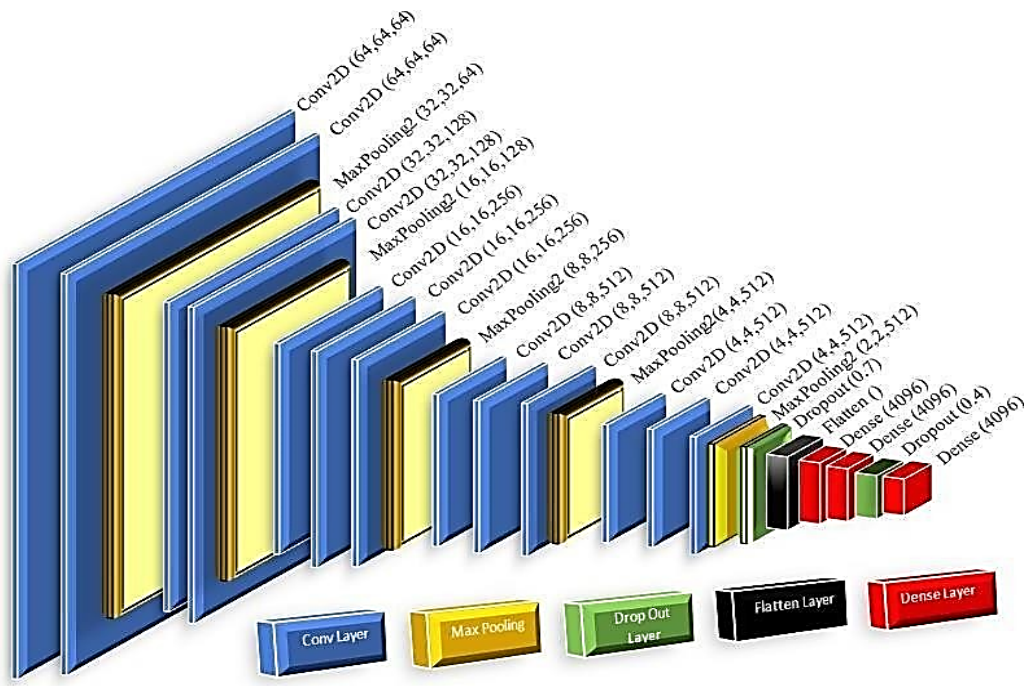

Figure 3.4: Convolutional Neural Network Model Structure

3.6 Training Process

The training process of the Convolutional Neural Network (CNN) for malware image classification was a pivotal aspect of this research, encompassing distinct stages to iteratively enhance the model's performance. Leveraging the TensorFlow framework, the training process unfolded over four carefully orchestrated stages, each characterized by a specific number of epochs.The following elucidates the methodology employed in these training stages:

3.6.1 Stage-wise Epoch Configurations

The training process was divided into four stages, each delineated by the number of epochs executed. The initial stage comprised a modest four epochs, providing an initial exploration of themodel's learning capacity.

Subsequent stages, with 14 epochs, 16 epochs, 20 epochs (incorporating L2 regularization),and 50 epochs (also with L2 regularization), were incrementally introduced. This staged approachfacilitated an iterative fine-tuning of the model across varying durations of training.

3.6.2 Integration of L2 Regularization

To address overfitting concerns that emerged during the initial stages of training, L2 regularization was introduced in the fourth and final stages, involving 20 and 50 epochs. This regularization technique-imposed constraints on the weights of the model, enhancing its ability togeneralize to unseen data. The strategic incorporation of L2 regularization was a responsive measure to mitigate the risk of the model fitting noise in the training dataset.

3.6.3 Autotune Functionality

The TensorFlow framework's autotune functionality played a crucial role in optimizing both the training and validation data pipelines. Autotuning automatically adjusts parameters, suchas the batch size and other configurations, to achieve optimal computational efficiency. This adaptive mechanism is particularly valuable when dealing with varying dataset sizes and complexities, as it ensures that the computational resources are utilized efficiently throughout the training process.

3.6.4 Iterative Monitoring and Optimization

Throughout the training process, various callback mechanisms were implemented to monitor and optimize the model's learning. These included callbacks for plotting loss curves, adjusting learning rates, early stopping to prevent overfitting, and model checkpointing to save the best-performing model weights. These callbacks collectively contributed to the iterative refinementof the CNN model.

3.7 Evaluation Metrics

3.7.1 Metrics Selection

The assessment of the CNN model's performance in classifying malware images transcended conventional metrics, employing a diverse set of custom evaluation measures. This section delves into the rationale behind selecting these metrics, each contributing uniquely to a holistic understanding of the model's capabilities.

3.7.2 Area Under the Curve (AUC) with ROC curve

A comprehensive measure of the model's ability to discriminate between classes, is particularly valuable for binary and multi-class classification tasks.

3.7.3 Precision

The ratio of correctly predicted positive observations to the total predicted positives quantifies the model's accuracy in identifying true positives.

3.7.4 Recall

The ratio of correctly predicted positive observations to the total actual positives, emphasizes the model's ability to capture all relevant instances.

3.7.5 Accuracy

The overall correctness of the model's predictions, providing a general performance indicator.

3.7.6 Specificity

A measure of the model's capacity to correctly identify negative instances, complementingrecall by focusing on true negatives.

3.7.7 Conditional Average Metric

An advanced metric that considers the average performance of the model across various conditions or subgroups, providing insights into class-specific behavior.

3.7.8 Sensitivity

A synonym for recall, emphasizing the model's responsiveness to positive instances.

3.7.9 F1 Score

The harmonic means of precision and recall, offer a balanced evaluation metric that considers both false positives and false negatives.

3.7.10 Holistic Understanding

These diverse metrics collectively provide a comprehensive understanding of the model's strengths and weaknesses. AUC and ROC curve analysis offers insights into the model's discriminatory power, while precision and recall shed light on its ability to minimize false positives and false negatives. Accuracy serves as a general performance gauge, and specificity ensures a balanced consideration of true negatives

3.7.11 Customization for Malware Classification

The selection of these metrics is specifically tailored to the nuances of malware image classification. The intricate nature of malware detection demands a multi-faceted evaluation approach that goes beyond simple accuracy, considering the consequences of both false positives and false negatives.

3.7.12 Practical Implications

By incorporating these diverse metrics, the evaluation process extends beyond a binary assessment of correctness. It provides a nuanced understanding of the model's performance, aidingin the identification of areas for improvement and facilitating informed decision-making in real- world deployment scenarios.

3.7.13 Model Compilation

The compilation of the Convolutional Neural Network (CNN) model is a critical step, in defining the model's behavior during training and evaluation. This section provides insights into the compilation process, detailing the choice of loss function and metrics tailored to the specific challenges posed by malware image classification.

3.7.14 Categorical Cross-Entropy Loss Function

The CNN model was compiled using the categorical cross-entropy loss function. This choice is apt for multi-class classification tasks, such as malware image classification, where the objective is to assign a single class label to each image among multiple possible categories. Categorical cross-entropy is well-suited for optimizing the model parameters to achieve accurate and confident predictions across all classes.

3.7.15 Specialized Metrics for Malware Classification

In addition to the loss function, a set of meticulously chosen metrics was specified during compilation. These metrics are designed to capture the nuances of malware classification, going beyond traditional metrics like accuracy. The inclusion of metrics such as precision, recall, specificity, and custom measures like AUC with ROC curve reflects a nuanced approach to evaluating the model's performance in the context of malware detection.

```
import tensorflow as tf
import matplotlib.pyplot as plt
import numpy as np

class_names = ['AgentTesla', 'Benign', 'CoinMinerXMRig', 'Danabot', 'Dridex', 'Formbook', 'Gh0stRAT', 'Glupteba', 'Gozi',
               'Heodo', 'NanoCore', 'Quakbot', 'RecordBreaker', 'RedLineStealer', 'Remcos', 'Tinba', 'Trickbot', 'Zeus']

num_images = len(validation_data)
for images, labels in validation_data.take(num_images):
    predictions = cnn_model.predict(images)
    for i in range(len(images)):
        predicted_class = np.argmax(predictions[i])

        # Map predicted class index to class label
        predicted_label = class_names[predicted_class]

        plt.imshow(images[i])
        plt.axis('off')
        plt.title('Predicted: {}'.format(predicted_label))
        plt.show()
```

Figure 3.5: Malware Prediction

3.8 Parameters and Hyperparameters

The training process involves careful tuning of various parameters and hyperparameters.

3.8.1 Learning Rate

This is a critical hyperparameter that sets the step size for each iteration as it approaches the loss function minimum. To modify the rate over epochs, one option is to employ a learning ratescheduler.

3.8.2 Batch Size

This indicates how many samples will propagate via the network in a single pass. To guarantee effective training without taxing memory capacity, a compromise is needed.

3.8.3 Epochs

The number of thorough runs through the training dataset is indicated by this. To provide adequate training without leading to overfitting, the number of epochs is specified.

3.8.4 Loss Function

Categorical Loss function are frequently employed as loss functions for regression tasks.

3.8.5 Regularization Techniques

To ensure that the model generalizes adequately to new data, overfitting is prevented by theuse of techniques like dropout or L2 regularization.

```
from tensorflow.keras.callbacks import LearningRateScheduler, ModelCheckpoint
from livelossplot.inputs.keras import PlotLossesCallback


def schedule(epoch, learning_rate):
    if epoch < 10:
        return learning_rate
    else:
        return learning_rate * tf.math.exp(-0.1)

lr_scheduler = LearningRateScheduler(schedule)
model_checkpoint = ModelCheckpoint("model_weights.h5", save_best_only=True, save_weights_only=True, monitor="val_loss")
from keras.callbacks import ModelCheckpoint,EarlyStopping
early_stop = EarlyStopping(monitor='val_accuracy',
                           patience=30,
                           restore_best_weights=True,
                           mode='max')

plot_loss_1 = PlotLossesCallback()
# modelcheck point save bast weights
```

Figure 3.6: Regularization Techniques

3.9 Important Libraries and Software Tools

In this research, Python has been utilized to facilitate preprocessing, analysis, and visualization of the final results. High-level programming languages like Python are versatile and known for being easy to read. Python is a programming language that supports a wide range of paradigms, including procedural, object-oriented, and functional programming. It also has a large standard library and a vibrant third-party package ecosystem. As a result, it has become a popular choice for a wide range of applications, including machine learning, artificial intelligence, web development, and data analysis.

Pandas and other libraries make it easier to get, clean, and effectively preprocess financial data. To produce educational visual aids illustrating past price trends and correlations, Matplotlib and Seaborn are widely used. Using machine learning frameworks such as TensorFlow, PyTorch, and Scikit-learn,one can create predictive models to analyze past data and forecast potential attacks in the future. Regressions and time series analysis are two statisticalanalyses in finance that are performed usingPython's stats model package. Python makes it simpleto integrate with financial APIs, giving usersaccess to sentiment analysis and real-time market data.

For the analysis and display of financial data, many PYTHON libraries are frequently used.The libraries used in this study include Pandas, Pandas_dataereader, NumPy, Matplotlib, TensorFlow, Keras, and JSON Library.

3.9.1 Pandas

Pandas is a versatile, open-source Python library focused on data analysis and manipulation.It offers essential data structures like Data Frames and Series to simplify the handling and analysisof structured data. Pandas are widely used in statistics, finance, and data science, streamlining tasks such as cleaning, converting, and examining datasets. It is an essentialtool for effectively workingwith structured data in the Python programming language, boasting features such as data alignment, fast handling of missing data, and powerful manipulation capabilities.

3.9.2 Keras

Keras is an open-source high-level neural network API that simplifies the process of creating, training, and deploying machine learning models. It is known for its easy-to-understand syntax and abstraction, which allows for the development of intricate neural network structures with minimal coding. Initially an independent library, Keras seamlessly integrated into TensorFlow, providing a user-friendly interface for building neural networks on the TensorFlow framework. Its popularity stems from its flexibility and ease of use, attracting both beginners and experienced professionals in the deep learning field.

3.9.3 TensorFlow

An open-source framework called TensorFlow is used to create and implement machine learning algorithms. Its scalability makes it possible to effectively divide computations among several CPUs and GPUs, improving performance. TensorFlow is easy to use and can be managed using a variety of computer languages, including Python and Java. Renowned for its intuitive interface, ease of use, and strong performance, TensorFlow has become one of the most widely used machine learning tools. Additionally, the integration of Keras, a high-level neural networks API, with TensorFlow provides a framework for developing other libraries.

3.9.4 NumPy

An open-source Python toolkit called NumPy is devoted to numerical operations and efficient handling of large, multi-dimensional arrays and matrices. NumPy is an essential tool for scientific computing and data processing since it provides an extensive set of functions for mathematical operations. NumPy is a vital tool for jobs involving linear algebra, statistical analysis,and mathematical computations. It is widely used in fields including machine learning,

data science, physics, and engineering. NumPy, well known for its effectiveness and adaptability,is a fundamental component of numerical computation in the Python environment.

3.10 Summary

This chapter outlines how the research design was structured for data collection and analysis, focusing on the methodologies and techniques chosen to meet specific research goals. It offers a detailed examination of the experimental methods utilized, aiming to provide a thorough understanding of how the study was conducted to achieve its objectives.

CHAPTER 4

RESULTS AND DISCUSSIONS

This chapter presents an in-depth analysis of the results obtained from the proposed study,focusing on the developed methodology. The chapter begins by detailing the performance metrics and outcomes, highlighting key findings and their implications for cybersecurity protocols. The chapter concludes with a comprehensive comparison of existing studies. This comparison evaluates the proposed approach against established methods, discussing improvements in accuracy, detection rates, and overall performance.

4.1 Overview

In machine learning and deep learning, model evaluation is an essential process that determines how well the trained model performs. The dataset is often split into training and testingsets. The model is trained on the training set and then assessed on the testing set to validate the performance evaluation of the proposed model. In this chapter, the results of a study on detecting malware attacks after training the model on 50 epochs are analyzed. The model is evaluated usingseveral measures including training and validation accuracy, F-1 Score, Recall, and sensitivity.

4.2 Accuracy

Figure 4.1 depicts the training process of a Convolutional Neural Network over 50 epochs.The smooth curve indicates a stable learning pattern. With a training accuracy of 0.96 and a validation accuracy of 0.95, the model demonstrates effective training and generalization.

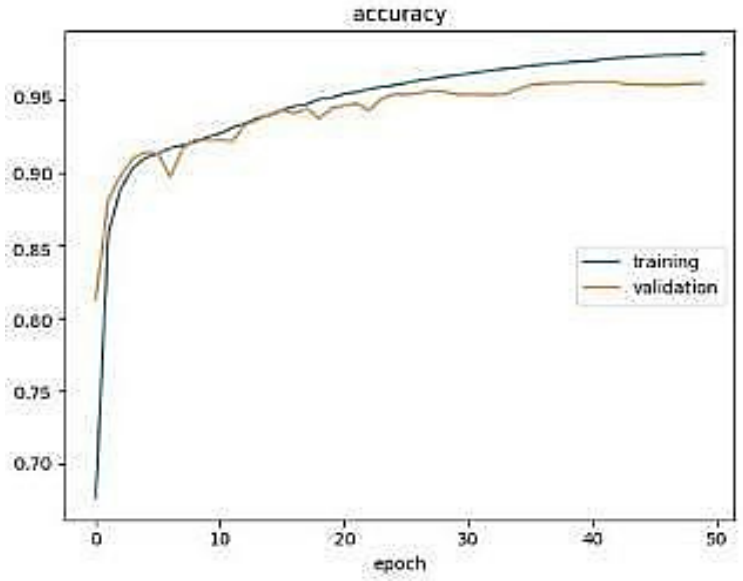

Figure 4.1: Accuracy

4.3 F1 Score

Figure 4.2 illustrates the performance of a model trained over 50 epochs, exhibiting remarkable smoothness. The training F1 score is 0.94, indicating a robust balance between precision and recall on the training set. Similarly, the validation F1 score at 0.93uggests the model'sability to generalize effectively to new data, with consistent and reliable performance across epochs

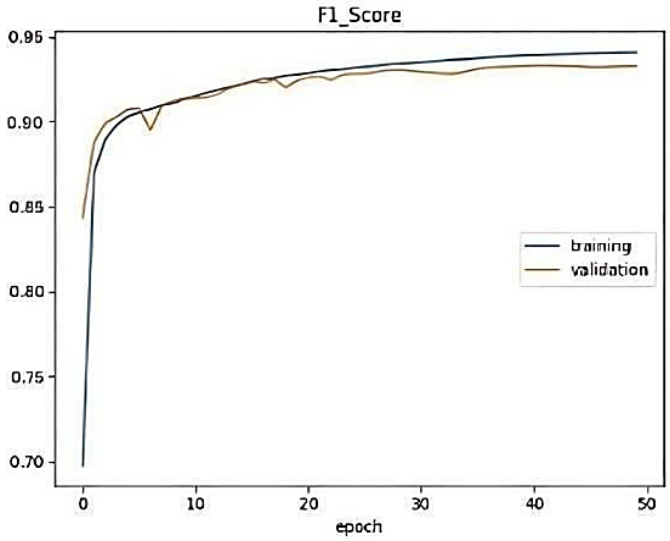

Figure 4.2: F1 Score

4.4 Conditional Average Metric

The Conditional Averaging algorithm displays a smoothly evolving graph during its training, reflecting a stable learning pattern in Figure 4.3. Notably, it achieves a high training accuracy of 0.89 and a commendable validation accuracy of 0.88. This consistency suggests the algorithm effectively balances its performance on seen and unseen data. The smoothness of the graph underscores the algorithm's robustness and reliable convergence during the training process.

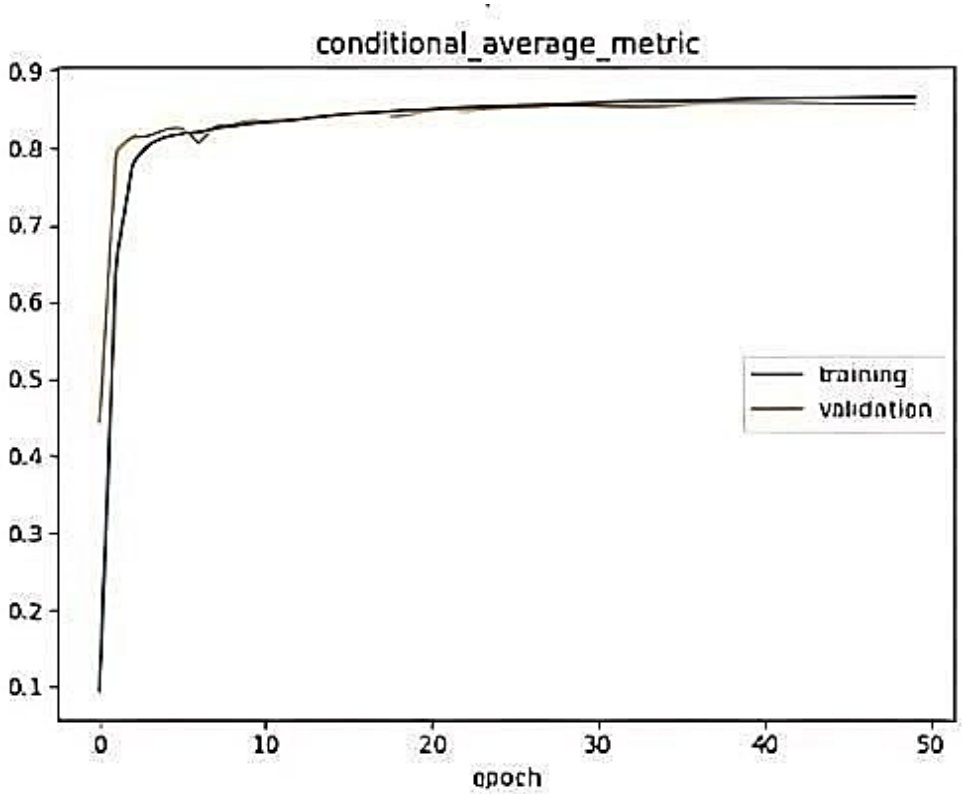

Figure 4.3: Conditional Average Metric

4.5 Loss

The loss graph demonstrates a significant reduction from an initial value of 20 at the first epoch to a minimal 0.03 by the 50th epoch. This remarkable decrease is attributed to both the effectiveness of the model's core architecture and the quality of the training data. Figure 4.4 illustrates the model's ability to continuously optimize its parameters, converging towards a state of minimal loss, indicative of a well-trained and finely tuned model.

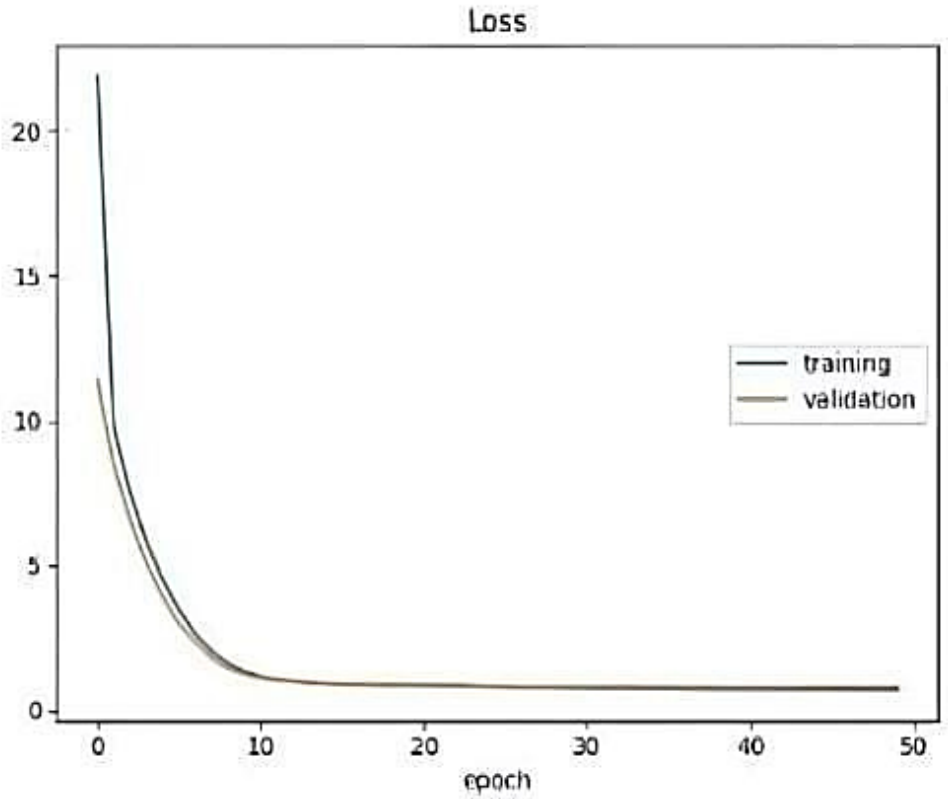

Figure 4.4: Loss

4.6 Precision

Figure 4.5 reveals a precision value of 0.95 for the training set and 0.94 for the validation set after 50 epochs. Precision, in this context, signifies the model's accuracy in correctly identifying positive instances among its predicted positives. The training precision of 0.95 indicates a high proportion of true positives among predicted positives during training. However, the slightly lower validation precision of 0.94 suggests a potential challenge in maintaining this accuracy on new and unseen data.

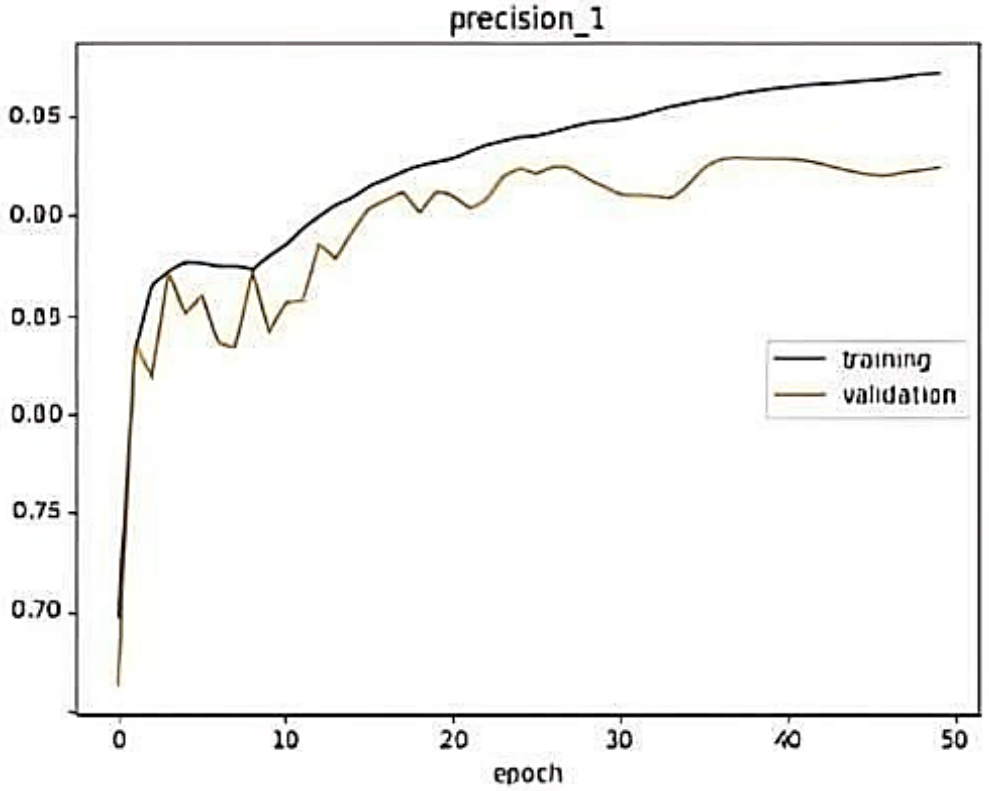

Figure 4.5: Precision

4.7 Recall

Figure 4.6 graph illustrates a training recall of 0.96 and a validation recall of 0.95 after 50 epochs. Recall, representing the model's ability to capture all positive instances, indicates that during training, the model identifies 96% of actual positives. The validation recall of 0.95 impliesa similar capability on new data. While the model shows a good ability to retrieve positive instances, there may be room for improvement, especially in generalizing to unseen data.

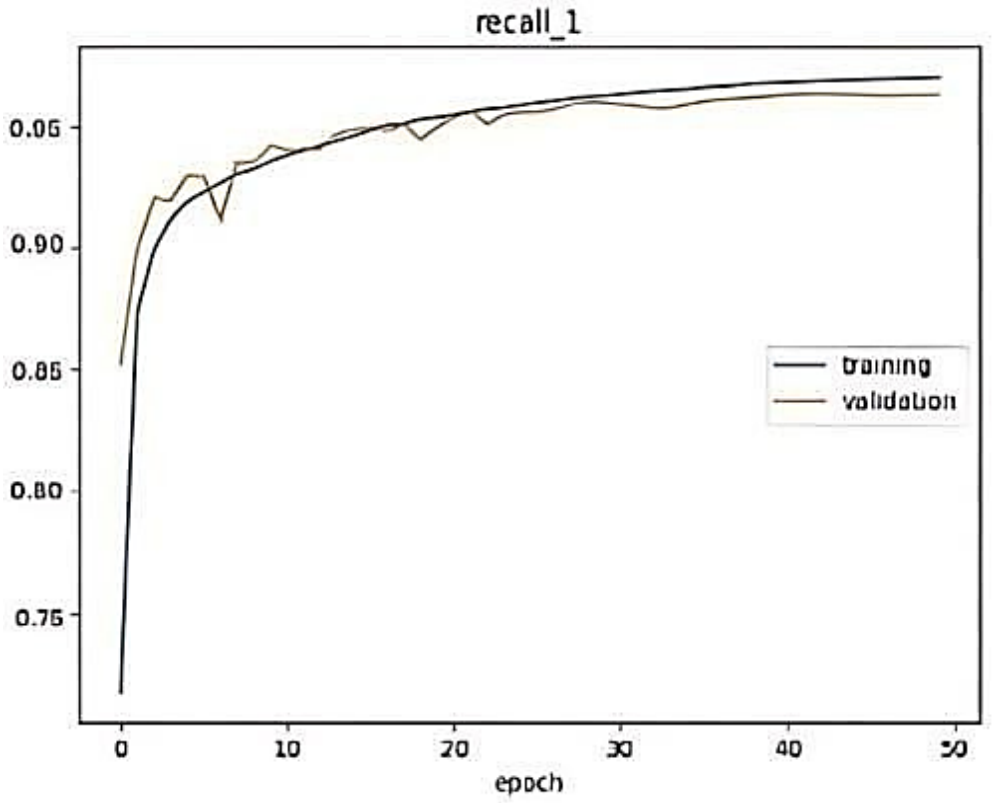

Figure 4.6: Recall

4.8 Sensitivity

Figure 4.7 indicates a training sensitivity of 0.96 and a validation sensitivity of 0.95 after 50 epochs. Sensitivity represents the model's capability to identify a high proportion of actual negative instances. The training sensitivity of 0.96 reveals the model's effectiveness in capturing the majority of true positives during training. With a validation sensitivity of 0.95, the model demonstrates consistent recall of new data, highlighting its ability to generalize well.

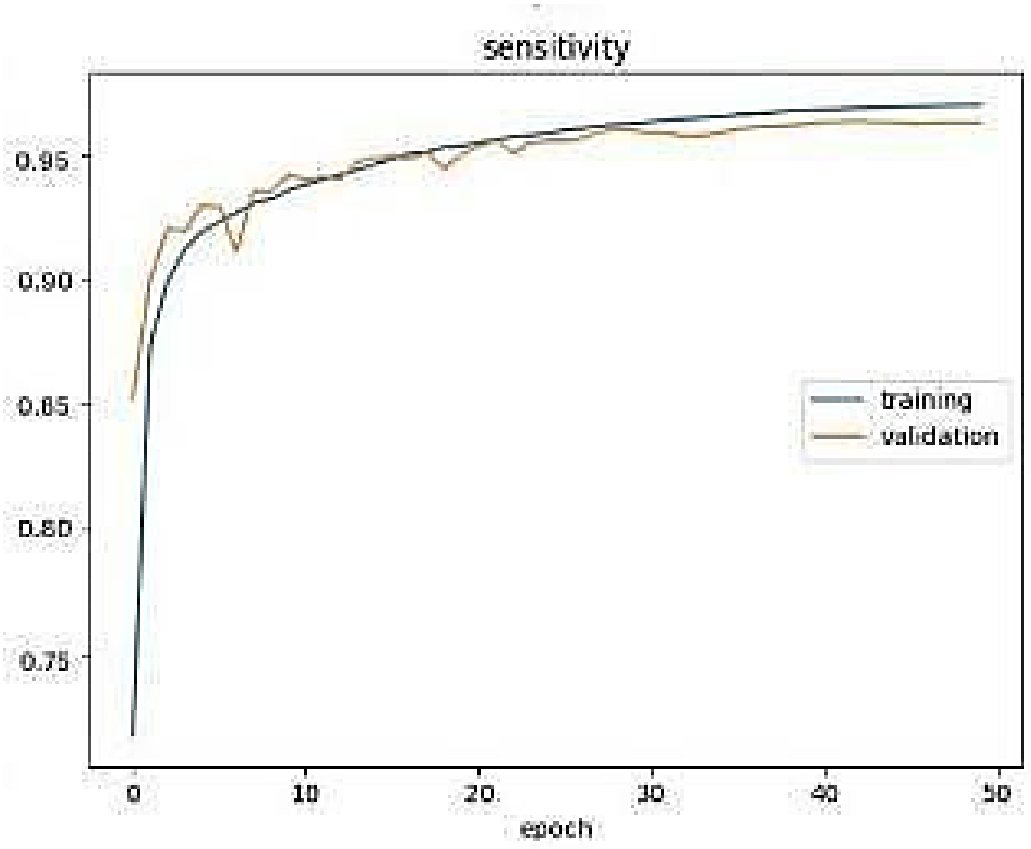

Figure 4.7: Sensitivity

4.9 Area Under Curve

The Area Under the Curve (AUC) graph exhibits an impressive performance with a trainingAUC of 0.99 and a validation AUC of 0.98 after 50 epochs shown in Figure 4.8. A high AUC value signifies excellent discrimination capability, indicating that the model can effectively distinguish between positive and negative instances. The close values between training and validation AUC suggest robust generalization, showcasing the model's ability to maintain discriminative power onunseen data.

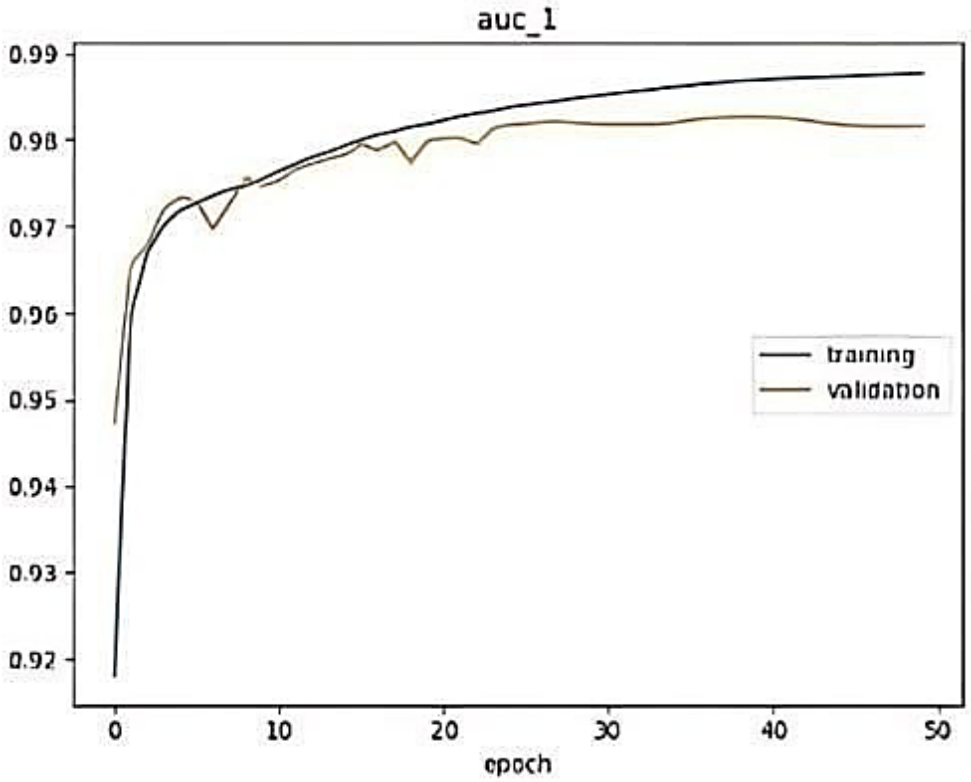

Figure 4.8: Area Under Curve

Table 4.1 presents a comprehensive visual summary of the performance metrics for the classification model, focusing on evaluation criteria: accuracy, F1 score, precision, recall, loss, conditional average metric, sensitivity, and area under the curve. The table provides a detailed viewof the model's performance. The concentration of higher values across all metrics suggests that themodel is accurate in its predictions.

Table 4.1: Experimental Results

Sr. No	Title	Training	Validation
1	Accuracy	96%	95%
2	Precision	96%	94%
3	Recall	96%	95%
4	F1 Score	94%	93%
5	Conditional Average Metric	89%	88%
6	Loss	10%	10%
7	Sensitivity	96%	95%
8	Area Under Curve	99%	98%

4.10 Comparison with Existing Studies

The study compares the proposed method with previous techniques to enhance the accuracyof cyber threat intelligence in onion services. Analysis using various metrics shows that the algorithm is highly accurate, outperforming previous machine and deep learning models. By leveraging a deep convolutional neural network, the study achieves superior accuracy in detectingmalware.

Table 4.2: Comparison with Existing Studies

Sr No	Author Name	Year	Methods	Accuracy
1	Varadharajan et al. [32]	2022	GA's	93%
2	Agarap [33]	2017	CNN-SVM GRU-SVM, MLP-SVM	85%
3	Zhong et al [34]	2021	Image Enhancement, CLAHE	96%
4	Moraliyage et al. [56]	2022	CNN and Grad-CAM	93%
5	Proposed Model	2024	CNN	96%

4.11 Discussions

The research focused on developing and employing a custom Convolutional Neural Network (CNN) designed for malware detection. This CNN, with 39,962,450 tunable parameters, was tailored to classify different types of malware, including notable examples such as AgentTesla, CoinMinerXMRig, and Dridex. During the initial phases of training, the model performed effectively in learning to recognize these specific malware types. The primary area of interest lies in the model's ability to process images and detect malware at any location within the image, ensuring comprehensive identification. This approach demonstrates the potential for applying advanced image processing techniques in the detection of malware, contributing to enhanced cybersecurity measures. However, the primary challenge encountered was overfitting. This issue arose when CNN excelled in detecting malware from the training set but failed to generalize its findings to new, unseen malware samples. Consequently, while the model demonstrated high accuracy with known threats, it struggled to maintain the same level of performance when confronted with unfamiliar malware. This limitation highlights the need for further refinement in the model to enhance its ability to recognize and handle novel malware

threatsthat were not included in the training data.

To tackle the issue of overfitting, several innovative techniques to enhance the model's performance. The primary strategy involved integrating Dropout into the model's architecture. Dropout is a regularization technique that randomly deactivates a subset of neurons during the training process. By doing so, it prevents the network from becoming overly reliant on any particular feature or set of features. This randomness encourages the model to learn more robust representations of the data, as it cannot depend on any single neuron or small group of neurons. Asa result, the risk of overfitting—where the model performs well on training data but poorly on new,unseen data—is significantly reduced. This approach helps in creating a more generalized model that can better handle variations in the input data and improve overall performance. ImplementingDropout enhances the model's ability to generalize and adapt to diverse and unseen data, which is crucial for effective malware detection.

In addition to integrating Dropout, an early stopping mechanism to further improve the model's performance. This technique involves monitoring the model's performance on a validationdataset throughout the training process. Training is automatically halted when the model's performance on this validation set ceases to improve. By stopping training at this optimal point, themodel continues to fit excessively to the training data.

This method helps avoid overfitting, wherethe model becomes too specialized to the training set and loses its ability to generalize to new, unseen data.

The early stopping mechanism thus enhances the model’s generalization capability, ensuring that it maintains a balance between learning from the training data and performing well on data it hasn’t encountered before. This approach is crucial for developing a robust model that can effectively handle various inputs and improve overall reliability in practical applications.

To enhance the model's performance, a learning rate scheduler is integrated into the approach. This scheduler dynamically adjusts the learning rate based on the model's performance metrics during training. By modifying the learning rate, the scheduler ensures that the model converges more smoothly toward an optimal solution. Specifically, it lowers the learning rate whenthe model shows signs of plateaus or when performance improvements slow down, allowing for finer adjustments in the learning process. Conversely, it may increase the learning rate if the modelis improving rapidly, speeding up the convergence. This dynamic

adjustment helps in maintainingan efficient learning pace, addressing issues such as overfitting by avoiding large, erratic updates that can destabilize the learning process. As a result, the learning rate scheduler contributes to morestable and consistent training, ultimately improving the model's ability to generalize well to new, unseen data. This technique ensures that the model not only fits the training data effectively but also retains its robustness in diverse real-world scenarios.

The model's accuracy on the malware dataset was 96%. Several criteria were utilized to assess accuracy, demonstrating how well the model worked: precision, recall, specificity, and F1 score. CNN model outperforms previous studies in terms of accuracy, precision, recall, F1 score, AUC, and loss, indicating that it is a highly effective model for the task at hand. The specific area of the image that malware comprises can vary depending on the type and characteristics of the malware. In the context of the CNN model described, the model is designed to detect and identify malware at any location within an image, regardless of the area it occupies. The detection process aims to ensure comprehensive identification, meaning that the model scans the entire image for potential malware, without a fixed or predetermined area of interest. This approach allows the model to dynamically pinpoint and classify malware, regardless of its position or size within the image. The deep architecture, regularization, and balanced metrics make it a strong performer compared to other models in the literature.

A simple CNN can outperform more complex hybrid models like CNN-SVM, GRU-SVM,or MLP-SVM in certain scenarios due to a few key factors. Firstly, CNNs are inherently designed for tasks like image recognition and pattern detection. Their architecture, with convolutional and pooling layers, efficiently captures spatial hierarchies and local dependencies in data, especially images. This allows CNNs to automatically learn features that are highly relevant to the task, making them very effective even in simpler architectures.

CNNs can also be considered better than Res-Net in situations where simplicity and computational efficiency are key. For smaller datasets or simpler tasks, CNNs provide faster training and inference without the complexity of residual connections. They are also easier to interpret and require fewer resources, making them suitable for less demanding applications. Additionally, CNNs may prevent overfitting more effectively in scenarios with limited data, as their simpler architecture avoids excessive parameterization.

On the other hand, hybrid models like CNN-SVM, GRU-SVM, or MLP-SVM combine

thefeature extraction power of deep learning models (CNN, GRU, MLP) with the classification strength of SVMs. While this combination can theoretically improve performance by leveraging the strengths of both techniques, in practice, these models can suffer from increased complexity. This added complexity may lead to longer training times, increased risk of overfitting, and higher computational resource requirements. Moreover, SVMs are not as efficient in handling large-scaledata or multi-class problems compared to neural networks, which are inherently more scalable andflexible for these tasks.

Thus, a simple CNN might outperform these complex approaches because it balances modelcomplexity and performance effectively. It learns directly from data in an end-to-end fashion, avoiding the need to combine separate models, which can introduce inefficiencies.

By meticulously adjusting the model architecture and integrating these advanced techniques, we achieved notable performance enhancements. The implementation of strategies such as Dropout, early stopping, and a learning rate scheduler has significantly refined the model'scapability. As a result, the model now exhibits improved generalization and accuracy in classifyingmalware images. These refinements have led to a more robust and reliable system, capable of better handling diverse and previously unseen data. This progress not only strengthens the effectiveness of our specific project but also has broader implications for the field of cybersecurity.

The advancements contribute valuable insights and methodologies that can be applied to similar challenges in cybersecurity, enhancing the overall quality and reliability of malware detection systems. By pushing the boundaries of what is possible with current technologies, these improvements offer a significant boost to the fight against cyber threats.

4.12 Summary

This chapter provides a comprehensive summary of the results obtained from the proposed study, focusing on the accuracy of the developed methodology. It details the performance metrics and key findings, highlighting their implications for enhancing cybersecurity protocols. Thechapter then presents a thorough comparison with existing studies, evaluating the proposedapproach against established methods in terms of accuracy, detection rates, and overall performance. This comparative analysis underscores the improvements and advancements made by the proposed method.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

Malware has increased in frequency in recent years, posing a global threat to individuals, companies, and digital assets. Even with the wide range of approaches and strategies put out for identifying and neutralizing malevolent agents, contemporary automated malware generation techniques continue to generate malware that is resistant to current detection technologies. The demand for sophisticated and precise malware detection tools has surged as a result. The study putforth a cutting-edge deep learning architecture for malware detection. To balance the dataset and lessen overfitting, firstly the model's generalization by data augmentation. The VG16 model was then employed as a feature extractor. Then, regularization strategies are used as a learning rate scheduler, model checkpoint, plot loss call back, and early stop to increase the suggested model's efficacy and accuracy. Consequently, the framework demonstrated exceptional performance in identifying various malware classes. Applying the suggested model to the Malware Image Datasetyielded the experimental findings. 125.004 PNG grayscale pictures that were taken from 17 malware families as well as innocuous applications make up the Malware Image collection. In thisstudy to find out how crucial the attention module is for malware detection. The model's accuracyon the malware dataset was 96%. Several criteria were utilized to assess accuracy, demonstrating how well the model worked: precision, recall, specificity, and F1 score. These findings show the efficient speed of the suggested model's operation while proving its efficacy in correctly categorizing malware. In subsequent research, the aim is to test various attention modules to determine which one is the most reliable and effective in terms of performance gains.

5.2 Future Work

Future work should focus on addressing the research gap in converting grayscale photographs into color images, and exploring advanced techniques and algorithms to enhance accuracy and efficiency in this area. Additionally, further research could investigate the applicationof deep learning frameworks in refining cybersecurity protocols, leveraging the insights gained from the generated results to develop more robust defenses against evolving malware threats. Thisentails exploring innovative approaches to integrate deep learning into real-time threat detection and mitigation strategies, thereby bolstering cybersecurity resilience in increasingly complex digital environments.

REFERENCES

[1] J. H. Li, "Cyber security meets artificial intelligence: A survey," Frontiers of Information Technology & Electronic Engineering, vol. 19, no. 12, pp. 1462–1474, 2018.

[2] R. Chetry and U. Sharma, "Dark web activity on Tor—Investigation challenges and retrieval of memory artifacts," in Proc. Int. Conf. Innovative Computing Communication.,pp. 953–964, 2021.

[3] P. Burda, C. Boot, and L. Allodi, "Characterizing the redundancy of DarkWeb. Onion services," in Proc. 14th Int. Conf. Availability, Reliability Security, pp. 1–10, Aug. 2019.

[4] H. Moraliyage, V. Sumanasena, D. De Silva, R. Nawaratne, L. Sun, and D. Alahakoon, "Multimodal classification of onion services for proactive cyber threat intelligence using explainable deep learning," IEEE Access, vol. 10, pp. 56044–56056, 2022.

[5] M. Vohra, A. Tiwari, P. Sharma, and A. Jayaswal, "Malware detection using deeplearning," Lecture Notes in Electrical Engineering, vol. 1079, no. 1, pp. 215–231, 2023.

[6] T. B. Lee, "#3 1982: The ARPANET community grows," Vox Conversations, Jun. 2, 2014. [Online]. Available: https://en.wikipedia.org/wiki/Internet.

[7] J. Strickland, "How stuff works: Who owns the internet?" Mar. 3, 2008. [Online]. Available: https://www.scribd.com/document/665313649/Internet.

[8] A Burnett, "What is the difference between the Surface Web, The Deep Web, and the Dark Web?" Pink Hat Technology Management. [Online]. Feb 20, 2020. Available: https://digitalcommons.law.uga.edu/cgi/viewcontent.cgi?article=1077&context=cle.

[9] J. Devine and F. Egger-Sider, "Beyond Google: The invisible web in the academic library,"

[10] J. Acad. Librarianship, vol. 30, no. 4, pp. 265–269, Aug. 2021.

[11] S. Shedden, "How do you want me to do it? Does it have to look like an accident? – An assassin selling a hit on the net; revealed inside the deep web," Sunday Mail, Jun. 8, 2014. [Online]. Available: https://www.sundaymail.co.uk/.

[12] A. Greenberg, "Hacker lexicon: What is the dark web?" Wired, Nov. 19, 2014. [Online]. Available: https://www.wired.com/.

[13] D. Schatz, R. Bashroush, and J. Wall, "Towards a more representative definition of cyber security," J. Digit. Forensics, Secur. Law, vol. 12, no. 2, 2017.

[14] M. Kianpour, S. Kowalski, and H. Øverby, "Systematically understanding cybersecurity economics: A survey," Sustainability, vol. 13, no. 24, p. 13677, 2021.

[15] S. Chaudhary, V. Gkioulos, and S. Katsikas, "Developing metrics to assess the effectiveness of cybersecurity awareness program," J. Cybersecurity, vol. 8, no. 1, pp. 1– 19, 2022.

[16] NCSC, "How cyber-attacks work," National Cyber Security Centre. [Online]. Available: https://www.ncsc.gov.uk/.

[17] A. K. U. Kamal and D. S. V, "Integrated threat intelligence platform for security operationsin organizations," Automatika, vol. 65, no. 2, pp. 401–409, 2024.

[18] C. E. Beckerman, "Is there a cyber security dilemma?" J. Cybersecurity, vol. 8, no. 1, pp.1–14, 2022.

[19] M. Al-Hawawreh and N. Moustafa, "Explainable deep learning for attack intelligence and combating cyber-physical attacks," Ad Hoc Netw., vol. 153, p. 103329, 2023.

[20] H. Wang, A. Singhal, and P. Liu, "Tackling imbalanced data in cybersecurity with transfer learning: A case with ROP payload detection," Cybersecurity, vol. 6, no. 1, 2023.

[21] K. Tongkachok, V. Samata, K. Nethravathi, M. Nirmal, L. R. Mohan, and Z. Z. Khan, "The empirical analysis of machine learning approaches for enhancing the cybersecurity for better quality," in Proc. 2nd Int. Conf. Innovative Practices Technol. Manag., pp. 95–100,2022.

[22] R. Tahir, "A study on malware and malware detection techniques," Int. J. Educ. Manag. Eng., vol. 8, no. 2, pp. 20–29, 2018. [Online]. Available: https://www.mecs-press.org/.

[23] S. Morgan, "Cybercrime to cost the world $10.5 trillion annually by 2025," Cybercrime Magazine, Nov. 13, 2020. [Online]. Available: https://cybersecurityventures.com/.

[24] B. Molina-Coronado, U. Mori, A. Mendiburu, and J. Miguel-Alonso, "Towards a fair comparison and realistic evaluation framework of android malware detectors based on static analysis and machine learning," Computers & Security, vol. 124, p. 102996, 2023.

[25] V Adam, "Computer virus," Encyclopedia Britannica. [Online]. Nov 24, 2024 Available: https://www.britannica.com/topic/cybercrime/Spam-steganography-and-e-mail-hacking.

[26] M. McDowell, "Understanding hidden threats: Rootkits and botnets," US-CERT, Mar. 29, 2017. [Online]. Available: https://www.us-cert.cisa.gov/.

[27] H. Gill, "Malware: Types, analysis and classifications," ResearchGate, Jun. 2022. [Online]. Available: https://www.researchgate.net/.

[28] "Proton Mac Trojan has Apple code signing signatures sold to customers for $50k," AppleInsider, Mar. 14, 2017. [Online]. Available: https://appleinsider.com/.

[29] "Trojan dropper," MalwareBytes, Jan. 30, 2020. [Online]. Available: https://www.malwarebytes.com/.

[30] R. Richardson and M. North, "Ransomware: Evolution, mitigation and prevention," Int. Manag. Rev., vol. 13, no. 1, pp. 10–21, 2017. [Online]. Available: https://www.imrjournal.org/.

[31] "State of malware," Malwarebytes. [Online]. Available: https://www.malwarebytes.com/.

[32] C. Paardekooper, N. Noman, R. Chiong, and V. Varadharajan, "Designing deep convolutional neural networks using a genetic algorithm for image-based malware classification," in Proc. 2022 IEEE Congr. Evolutionary Computation (CEC), pp. 1–10, 2022.

[33] A. F. Agarap, "Towards building an intelligent anti-malware system: A deep learning approach using support vector machine (SVM) for malware classification," arXiv, Dec. 2017. [Online]. Available: https://arxiv.org/.

[34] F. Zhong, Z. Chen, M. Xu, G. Zhang, D. Yu, and X. Cheng, "Malware-on-the-brain: Illuminating malware byte codes with images for malware classification," arXiv, Aug. 2021. [Online]. Available: https://arxiv.org/.

[35] D. Lee, "Darknet raids were 'overblown' by police, says Tor Project," BBC, Nov. 10, 2014. [Online]. Available: https://www.bbc.com/.

[36] E. Eaton, S. Sasy, and I. Goldberg, "Improving the privacy of Tor onion services," Lecture Notes in Computer Science, vol. 13269, pp. 273–292, 2022.

[37] J. Bergman and O. B. Popov, "Recognition of Tor malware and onion services," J. Comput. Virol. Hacking Tech., 2023.

[38] B. Huang and Y. Du, "Discovering onion services through circuit fingerprinting attacks,"in Proc. 2022 IEEE/ACM 7th Symp. Edge Comput. (SEC), pp. 498–503, 2022.

[39] J. Ödén, "Deanonymizing onion services by introducing packet delay," 2022. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1664834/FULLTEXT02.

[40] C. Wang, J. Luo, Z. Ling, L. Luo, and X. Fu, "A comprehensive and long-term evaluationof the Tor network: performance, security, and hidden services," Future Internet, vol. 15, no. 1, pp. 1–16, 2023.

[41] S. Misra and D. Dhillon, "Detecting malicious onion services using a hybrid approach," in Proc. 19th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom), pp. 755–762, 2020.

[42] K. Omon, S. Amir, and R. Kumari, "Understanding the dark web," FBI. January 2021

[43] [Online]Available: https://www.researchgate.net/publication/348598990_Understanding_the_Dark_Web.

[44] "What is the difference between the Surface Web, The Deep Web, and the Dark Web?" Pink Hat Technology Management. [Online]. Available: https://www.pinkhattech.com/.

[45] T. G. Lewis, Critical Infrastructure Protection in Homeland Security: Defending a Networked Nation, 2nd ed., John Wiley & Sons, Inc., 2021.

[46] J. H. Allen, The CERT Guide to System and Network Security Practices, Addison-Wesley, 2001.

[47] Internet security threat report, Symantec Corporation, 2023. [Online] Available: https://docs.broadcom.com/doc/istr-03-jan-en.

[48] Gartner, Inc., Magic Quadrant for endpoint protection platforms. [Online]. Available: https://www.gartner.com/.

[49] Global risks report, World Economic Forum, 2024. [Online]. Available: https://www.weforum.org/.

[50] NIST cybersecurity framework, National Institute of Standards and Technology (NIST), 2022. [Online]. Available: https://www.nist.gov/.

[51] International Organization for Standardization (ISO), ISO/IEC 27001:2022 – Information technology – Security techniques – Information security management systems – Requirements, 2023. [Online]. Available: https://www.iso.org/.

[52] K. S. Sangher, A. Singh, H. M. Pandey, and V. Kumar, "Towards safe cyber practices: Developing a proactive cyber-threat intelligence system for Dark Web forum content by identifying cybercrimes," Information, vol. 14, no. 6, 2023.

[53] K. Ahmed, S. K. Khurshid, and S. Hina, "CyberEntRel: Joint extraction of cyber entities and relations using deep learning," Computers & Security, vol. 136, p. 103579, 2023.

[54] M. Dekker and L. Alevizos, "A threat-intelligence-driven methodology to incorporate uncertainty in cyber risk analysis and enhance decision-making," Security & Privacy, pp. 1–18, 2023.

[55] N. Sun, M. Ding, J. Jiang, et al., "Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives," IEEE Commun. Surveys & Tutorials, vol. 25, no. 3, pp. 1748–1774, 2023.

[56] S. Samtani, H. Chen, M. Kantarcioglu, and B. Thuraisingham, "Explainable artificial intelligence for cyber threat intelligence (XAI-CTI)," IEEE Trans. Dependable Secure

Computer., vol. 19, no. 4, pp. 2149–2150, 2022.

[57] H. Moraliyage, V. Sumanasena, D. De Silva, R. Nawaratne, L. Sun, and D. Alahakoon, "Multimodal classification of onion services for proactive cyber threat intelligence using explainable deep learning," IEEE Access, vol. 10, pp. 56044–56056, 2022.

[58] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF)feature selection technique," Cybersecurity, vol. 5, no. 1, pp. 1–22, 2022.

[59] F. A. Ghaleb, M. Alsaedi, F. Saeed, J. Ahmad, and M. Alasli, "Model using ensemble learning," Sensors, pp. 1–20, 2022.

[60] Q. Li, D. Tan, X. Ge, H. Wang, Z. Li, and J. Liu, "Understanding security risks of embedded devices through fine-grained firmware fingerprinting," IEEE Trans. Dependable Secure Computer., vol. 19, no. 6, pp. 4099–4112, 2022.

[61] B. Herron and J. Hally, "Implementing security controls to IoT wireless technologies," GIAC Gold Certification, 2022.

[62] F. Aldauiji, O. Batarfi, and M. Bayousef, "Utilizing cyber threat hunting techniques to find ransomware attacks: A survey of the state of the art," IEEE Access, vol. 10, pp. 61695–61706, 2022.

[63] A.-Ofori S., S. Islam, S. W. Lee, et al., "Cyber threat predictive analytics for improving cyber supply chain security," IEEE Access, vol. 9, pp. 94318–94337, 2021.

[64] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF)feature selection technique," Cybersecurity, vol. 5, no. 1, pp. 1–22, 2022.

[65] T. Holler, T. Raab, M. Roland, and R. Mayrhofer, "On the feasibility of short-lived dynamic onion services," in Proc. 2021 IEEE Symp. Security Privacy Workshops (SPW 2021), pp. 25–30, 2021.

[66] A. Tudu, D. Bainbridge, and B. Rogers, "Finding a safe port: Cyber-security analysis for open source digital library software," in Proc. ACM/IEEE Joint Conf. Digital Libraries, pp. 349–352, 2020.

[67] M. A. Albahar, R. A. Al-Falluji, and M. Binsawad, "An empirical comparison onmalicious activity detection using different neural network-based models," IEEE Access, vol. 8, pp. 61549–61564, 2020.

[68] Z. N. Zarandi and I. Sharifi, "Detection and identification of cyber-attacks in cyber- physical

systems based on machine learning methods," in Proc. 2020 11th Int. Conf. Information Knowledge Technology (IKT 2020), pp. 107–112, 2020.

[69] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: Visualization and automatic classification," in Proc. ACM Int. Conf. Advanced ComputingScience, July 2014.