UNIVERSIDAD DE DEUSTO

THE ART OF CYBER THREAT HUNTING: HARNESSING AI FOR ADDRESSING NEWFANGLED CYBERSECURITY CHALLENGES

ALBERTO MIRANDA GARCÍA

Bilbao, 12 de junio de 2024

UNIVERSIDAD DE DEUSTO

THE ART OF CYBER THREAT HUNTING: HARNESSING AI FOR ADDRESSING NEWFANGLED CYBERSECURITY CHALLENGES

Doctoral thesis presented by Alberto Miranda García
within the Doctoral Program in Engineering for the Information Society and Sustainable Development

Supervised by
Dr. Iker Pastor López
and
Dr. Pablo Garcia Bringas

The PhD.	Co-supervisor	Co-supervisor
Firmado digitalmente por ALBERTO MIRANDA GARCIA Fecha: 2024.11.04 10:45:52 +01'00'	Firmado digitalmente por GARCIA BRINGAS, PABLO (FIRMA) Nombre de reconocimiento (DN): c=ES, serialNumber=22754794C, sn=GARCIA, givenName=PABLO, cn=GARCIA BRINGAS, PABLO (FIRMA) Fecha: 2024.11.04 11:31:33 +01'00'	PASTOR LOPEZ IKER - 78928122B Firmado digitalmente por PASTOR LOPEZ IKER - 78928122B Fecha: 2024.11.04 11:58:25 +01'00'

Bilbao, 12 de junio de 2024

Alberto Miranda-García: *The Art of Cyber Threat Hunting: Harnessing AI for Addressing Newfangled Cybersecurity Challenges*©, June 2024.

Website: https://albertomgv.com

Email: miranda.alberto@deusto.es

A mi familia y amigos, por su apoyo incondicional.

Abstract

Nowadays, cyber security has become an important issue, as it has a direct impact on all sectors of our society. The growth of digital technologies in key sectors such as public administration, healthcare and finance, highlights the urgent need for strong cyber defences. The importance of cyber security is becoming increasingly apparent as we move forward in this digital age, stating that it is a crucial component in protecting our digital lives in the face of a quickly changing cyber environment.

Along with the rapid evolution of technology, the cyber threat landscape has expanded, posing new challenges that require innovative solutions and introducing new risks and vulnerabilities. This evolution highlights a paradox where each technological milestone potentially opens up new vulnerabilities. In this context, the dynamic nature of cyber risks becomes evident, calling for a shift towards more adaptive and sophisticated cyber security measures to stay ahead of potential threats.

In response to these evolving cyber threats, the scientific community has intensified its focus on cybersecurity research, employing a mix of traditional and innovative methodologies. As a result, the shift towards integrating artificial intelligence (AI) into cybersecurity strategies has been driven. The integration of artificial intelligence (AI) into cybersecurity marks an important turning point in the fight against cyber threats by harnessing the potential of AI to identify, mitigate and prevent threats.

The core of this thesis focuses on the exploitation of AI to improve cyber threat hunting methodologies. It presents a series of new methodologies, integrating AI techniques to address the 3 critical cybersecurity challenges, with the aim of raising the efficiency and effectiveness of cyber threat hunting. The research addresses current cybersecurity challenges and anticipates future threats in a rapidly evolving cyber landscape. The approach presented here is not merely an academic exercise, but a practical framework designed to address real-world cybersecurity challenges through the application of AI technologies, setting a new standard for future research and practice in this field.

The practical implications of this thesis are extensive and go beyond theoretical contributions to offer real solutions to cyber challenges. Through a series of experiments, this research demonstrates how AI can be effectively applied in the field of cybersecurity, providing new and efficient methodologies. The results not only highlight the feasibility of AI to improve cybersecurity efforts, but also pave the way for future research and work in this field.

This research aims to bridge the gap between AI and cybersecurity, with the goal of significantly improving the efficiency and effectiveness of cyberthreat hunting by providing new methodologies and technics. This thesis, as suggested by its name 'The Art of Cyber Threat Hunting: Harnessing AI for Addressing Newfangled Cybersecurity Challenges', aims to pioneer a new frontier in cybersecurity methodologies and highlights the powerful role of AI as a transformative tool for identifying, analysing and addressing cyber threats.

Throughout the thesis, the 3 most prominent issues in the world of cybersecurity are analysed and addressed. The conclusions obtained in this thesis not only validate the hypothesis of the effectiveness of AI in cybersecurity, but also show new techniques and methodologies, which imply an advance for future research.

Agradecimientos

Antes de comenzar a exponer el trabajo llevado a cabo en esta tesis doctoral, creo que es importante agradecer a todas aquellas personas que me han ayudado a lograr este objetivo.

Esta tesis doctoral es fruto de la orientación de mis directores, Iker Pastor y Pablo García. Su confianza y la oportunidad de entrar en el grupo Deusto4Knowledge y su constante apoyo a lo largo de este proceso han sido cruciales. Me han guiado con conocimiento, ayudando a crecer en mi carrera como investigador y permitiéndome alcanzar este objetivo. A ellos, tambien quiero agradecer eternamente a la Universidad de Deusto por darme esta oportunidad y apoyarme a lo largo de este camino.

Quiero dar las gracias a cada uno de los miembros de mi equipo en DeustoTech, quienes han sido compañeros en este camino académico. Su apoyo, tanto profesional como personal, ha sido fundamental. Nerea, Asier, Ana, Erik, Unai, Guillermo, y Joseba no solo me han ayudado en cada desafío que se ha presentado, sino que también han contribuido a crear un ambiente de colaboración y aprendizaje mutuo. Su disposición incondicional para ayudar, compartir conocimientos y apoyo en los momentos más difíciles ha permitido terminar mi experiencia doctoral.

Especialmente quiero agradecerle a Alberto Fernandez por haberme acompañado no solo a lo largo de mi trabajo de tesis, sino a lo largo de todos estos años. Muchas gracias por apoyarme y compartir conmigo todos los momentos buenos así como todos los malos que hemos tenido que vivir. Realmente, siempre has sido una fuente de inspiración y apoyo para mí.

Sobre todo quiero agradecer este trabajo a mi familia, el pilar fundamental sobre el que se sostiene esta tesis doctoral. Su constante ánimo y apoyo incondicional en cada etapa de este viaje han sido esenciales para mí. En los momentos de mayor tensión y cansancio, cuando la saturación de la vida académica parecía sobrepasarme, su comprensión y paciencia han sido mi refugio. Gracias a ellos, pude encontrar la fuerza necesaria para continuar, recordándome la importancia de la perseverancia. Por estar siempre ahí, por sus palabras de ánimo y por soportarme en mis momentos más difíciles, les estoy eternamente agradecido.

Todo este trabajo tampoco hubiese sido posible sin mi segunda familia, que son mis amigos. Quiero agradecer enormemente a mi *cuadrilla*, por haberme apoyado a lo largo de este trabajo y de toda mi vida, teniendo mucha confianza en mí y animandome a cumplir todo aquello con lo que sueño. Agradecer tambien a todas las personas de *Verano Azul*, que de una manera u otra siempre han estado ahi, me han ayudado a seguir adelante y son unas personas muy importantes para mi. Y especialmente a mi círculo de *León*, agradecer a todas esa personas que forman parte de mi vida, que me apoyan incondicionalmente y han dado luz a mi vida.

Y por último, tambien quiero dar las gracias a todos los que habéis hecho posible esta tesis y no estáis en estos agradecimientos.

Alberto Miranda Garcia

Contents

List of Figures

List of Tables

Introduction 1

1.1 Motivation

In today's digital age, our dependence on technology has become intertwined with our daily lives. Organisations in all sectors, from finance to healthcare, rely heavily on cyber systems to run their daily operations. However, this growing reliance carries significant risks, as criminals find increasingly sophisticated methods to commit cybercrimes. The motivation behind this thesis is rooted in the critical need to address the disparity between rapid advances in malicious cyber activities and traditional cybersecurity methodologies that struggle to keep up.

Cybersecurity is not just about protecting data; it is about safeguarding critical infrastructure, protecting personal privacy, maintaining public trust, and in some cases, ensuring national security. With notable incidents such as the WannaCry ransomware attack in 2017 [50], which affected more than 200,000 computers in 150 countries, and the more recent SolarWinds hack in 2020 [7], which compromised thousands of organisations globally, it is painfully clear that current cybersecurity tactics are in need of evolution. These incidents not only caused substantial financial losses, but also damaged the public's confidence in the ability of institutions to protect their data [214].

This is where artificial intelligence (AI) comes in. AI has the potential to transform the field of cybersecurity by addressing its most pressing challenges: rapid threat detection, real-time incident response, and mitigating the cybersecurity talent shortage, to name a few. With the ability to rapidly analyse vast amounts of data and detect anomalies, AI can not only identify known threats, but is also incredibly effective at uncovering zero-day attacks and advanced evasion techniques that defy conventional methods.

In addition, AI can alleviate the growing skills gap in the cybersecurity field. According to Cybersecurity Ventures, there are expected to be 3.5 million unfilled cybersecurity jobs by 2025 [261]. AI tools can automate repetitive, low-level

This thesis explores the urgent need to evolve cybersecurity methods in response to advanced cybercrime. It then focuses on artificial intelligence (AI) as a transformative solution for cybersecurity. The potential of AI lies in rapid threat detection, real-time incident response and the nature of cybersecurity risk types. The thesis aims to explore the role of AI in evolving cybersecurity from a defensive intelligence system to a proactive and predictive one.

tasks, allowing cybersecurity personnel to focus on more strategic, higher-level activities.

However, despite its potential, the application of AI in cybersecurity is still in its early stages and faces challenges in terms of public understanding, trust, and regulatory frameworks. This thesis aims to explore in depth how AI can be a game changer in cybersecurity, not only as a reactive tool, but as a proactive and predictive system that transforms cybersecurity from a line of defence to one of intelligence.

The motivation for this research stems from the urgent need to advance our cyber defences, the promise that AI offers in this field, and the desire to explore, test and better understand how AI can, and will, revolutionise the way we protect our cyber systems. With growing and changing threats, it is imperative that we adapt and arm ourselves with technology that not only combats current threats but also anticipates and evolves with future ones. AI is not just a tool; it represents a new frontier in our collective fight against cybercrime.

1.2 Current State of Cybersecurity and Its Challenges

The digital revolution, while opening an era of connectivity and technological advances, has also exposed individuals, businesses and nations to cybersecurity threats. As our reliance on digital systems has grown, so too has our vulnerability to cyber-attacks. The landscape of cybersecurity is a battlefield that's evolving with astonishing speed, making it a challenging domain that's both dynamic and complex.

Table 1.1: The percentage that Cybercrime represents of the total criminal offenses in Spain.

Year	Percentage (%)
2018	7,5%
2019	9,9%
2020	16,3%
2021	15,6%
2022	16,1%

The importance of cybercrime is currently growing year by year, as shown by the increase in the number of known incidents and their proportional weight in the overall crime shown by Spanish Government. It can be observed in Table 1.1 that we have gone from 7.5% in 2018 to 16.1% in 2022 [82].

During the period from 2018 to 2022, there is an increase in computer crimes. In this way, we can see that in 2022, a total of 374,737 incidents were reported (Figure 1.1), which represents a 22.7% increase compared to the previous year.

Out of this figure, 89.7% corresponds to computer fraud (scams), and 4.3% to threats and coercion.

HECHOS CONOCIDOS	ene	feb	mar	abr	may	jun	jul	ago	sep	oct	nov	dic	TOTAL
ACCESO E INTERCEPTACIÓN ILÍCITA	307	367	426	420	538	502	529	466	575	457	570	421	**5.578**
AMENAZAS Y COACCIONES	1.326	1.434	1.570	1.282	1.319	1.403	1.322	1.340	1.288	1.295	1.255	1.148	**15.982**
CONTRA EL HONOR	78	104	127	91	91	107	107	101	114	103	90	78	**1.191**
CONTRA PROPIEDAD INDUST./INTELEC.	9	12	16	9	1	13	5	13	12	10	6	8	**114**
DELITOS SEXUALES(*)	168	135	195	119	159	145	90	128	138	139	137	93	**1.646**
FALSIFICACIÓN INFORMÁTICA	910	1.075	1.239	1.045	1.200	1.074	962	925	1.045	1.014	1.143	937	**12.569**
FRAUDE INFORMÁTICO	31.873	25.471	27.545	25.878	26.147	25.026	25.582	28.681	31.973	30.890	29.166	27.763	**335.995**
INTERFERENCIA DATOS Y EN SISTEMA	127	126	134	155	147	137	134	155	132	128	124	163	**1.662**
Total HECHOS CONOCIDOS	**34.798**	**28.724**	**31.252**	**28.999**	**29.602**	**28.407**	**28.731**	**31.809**	**35.277**	**34.036**	**32.491**	**30.611**	**374.737**

Figure 1.1: Monthly distribution of known crimes in the year 2022 reported by the Spanish Ministry of Interior [82].

1.2.1 Types of Cyber Threats

In today's interconnected world, the landscape of cyber threats is not only vast but also highly complex, consisting of a multitude of attacks that exploit various vulnerabilities within digital systems. These threats are no longer the pranks of lone hackers, but sophisticated strategies employed by organized crime groups and state-sponsored actors aiming at financial gain, espionage, and disruption of critical infrastructure. As our reliance on digital technology soars, the potential impact of these cyber threats magnifies, necessitating a comprehensive understanding of the diverse types of cyberattacks.

In 2017, about 42% of the cyber crimes reported were due to issues with non-payment or non-delivery. This mainly involved fraud in online shopping, where goods were paid for but not delivered, and instances where promised payments were not made. Additionally, 28% of the cyber crimes involved personal data breaches and phishing scams. Other forms of cyber attacks like identity theft and credit card fraud were less common Figure 1.2.

By 2022, phishing emerged as the dominant form of cyber attack. Over the past year, it accounted for more than half of all online criminal activities. Although email phishing has been prevalent since the early days of the internet, hackers have now developed more sophisticated phishing techniques, adapting them to various online platforms.

This section goes into specific categories of cyberthreats, from malware and phishing to ransomware and zero-day exploits, dissecting their mechanisms, identifying their targets and assessing their impact on individuals and businesses.

It is important to highlight the complex and changing landscape of cyber threats in the digital age, emphasising their transition from the work of lone hackers to the sophisticated strategies of organised crime and state actors. The section aims to provide a comprehensive understanding of various cyber-attacks, including malware, ransomware and zero-day exploits, focusing on their mechanisms, targets and impacts on individuals and businesses.

Phishing: Phishing attacks are a prime example of social engineering tactics, where attackers craft seemingly legitimate communications to elicit confidential information from unsuspecting victims. These deceptive appeals can vary from emails mimicking customer service requests to sophisticated spear-phishing campaigns targeting specific individuals within an organization [296]. Often, these attacks harness the familiarity of recognized brands or institutions to gain trust. Technical countermeasures against phishing include email filtering, domain authentication protocols like DMARC[1], and user training to recognize and report suspicious communications.

1: DMARC is an email authentication mechanism. It is designed to give email domain owners the ability to protect their domain from unauthorised use, commonly known as email spoofing.

Ransomware: Ransomware represents a dire threat model where attackers encrypt critical data and demand payment for its release. These attacks capitalize on various entry points, from phishing emails to exploiting network vulnerabilities. The encryption is often robust, leaving victims with limited options—pay the ransom, hope for a decryptor, or restore from backups, which is the recommended preparation strategy. Ransomware's impact is amplified by its ability to spread laterally within networks and the increasing use of cryptocurrencies to facilitate anonymous transactions.

Zero-Day Attacks: Zero-day attacks exploit vulnerabilities for which there is no current fix, allowing attackers to infiltrate systems undetected. These vulnerabilities are goldmines for attackers as they can be exploited until a patch is developed and applied. The technical complexity of defending against zero-day exploits lies in their very nature—they are unknown to security professionals at the time of exploita-

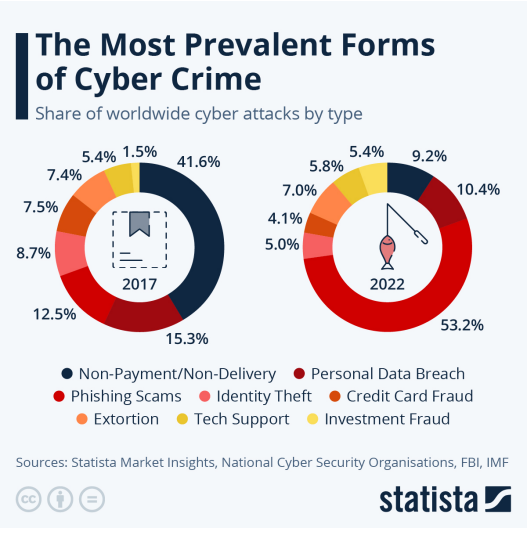


Figure 1.2: Leading Types of Cyber Crime. Global Distribution of Cyber Attack Categories. A comparison between 2017 and 2022. [289].

tion. Strategies to mitigate these attacks include deploying intrusion detection systems (IDS) that monitor for suspicious activity indicative of unknown exploits and adopting a robust patch management process.

Advanced Persistent Threats (APTs): APTs are sophisticated, stealthy, and strategic in nature. Unlike other attacks, APTs establish a long-term presence on a network, aiming to steal sensitive data over time. They often involve complex multi-vector attack strategies, lateral movement within networks [110], and stealthy exfiltration of data. Detection and mitigation require comprehensive monitoring, advanced anomaly detection algorithms, and a layered defense strategy that secures potential entry points.

Insider Threats: Insider threats manifest from within an organization, often harder to detect due to the legitimate access insiders have. Technical indicators of insider threats include abnormal access patterns or data transfer volumes. Mitigation strategies involve a combination of least privilege access controls, user behavior analytics (UBA), and comprehensive logging and monitoring of user activities.

Emerging Threats: The cybersecurity landscape is continuously evolving with emergent threats like AI-powered cyber attacks that adapt to defensive measures, crypto-jacking that secretly uses computer resources to mine cryptocurrency, and IoT-based attacks exploiting the vulnerabilities of a rapidly expanding network of connected devices. These next-generation threats can be more insidious and less detectable, requiring sophisticated AI-driven defenses and a proactive, rather than reactive, security posture.

Each type of threat requires a nuanced understanding of its construction, deployment, and potential damage. As such, the arsenal to combat these threats is diverse, including technical solutions, policy measures, and user education. The cybersecurity community is in a perpetual arms race with threat actors, and this section endeavors to dissect and understand the current state of this dynamic confrontation.

1.2.2 Cybersecurity Skills Gap

The cybersecurity skills gap remains a significant issue as we move through 2023, with the Future of Jobs report highlight-

ing cybersecurity as one of the top strategically emphasized skills needed in the workforce. Despite this, there is a shortage of 3.4 million cybersecurity experts required to support the global economy, a figure that is set to increase as emerging technologies become more prevalent[86]. This gap is particularly critical in sectors such as electricity, payments, and hospitals, which face the largest shortages in skilled cybersecurity professionals and are thus highly vulnerable to cyberattacks.

Although the cybersecurity workforce has grown to 5.5 million, the gap of unfilled positions is widening at an alarming rate, exacerbated by the changing nature of cyber threats and the need to continually update skills. Organisations are struggling to fill critical positions in the face of ever-increasing cyber threats and breaches. The gap not only means increased vulnerability to attacks, but also increased costs and economic impact due to potential breaches.

The cybersecurity workforce has indeed grown to 5.5 million people, marking an 8.7% increase over the past year. Yet, this growth is overshadowed by the rate at which the workforce gap is expanding, with an alarming 4 million unfilled positions worldwide, representing a 12.6% increase in just one year[277]. This shortage is exacerbated by the fast-paced nature of cyber threats which necessitates continuous learning, adaptation, and staying updated with emerging technologies, compliance requirements, and best practices[122].

Organizations are struggling not only to fill these critical roles but also to cope with the increased frequency and sophistication of cyber threats and breaches[85]. This situation is unlikely to improve shortly, as the need for skilled professionals continues to outpace the availability of such talent[270]. The focus for organizations in addressing this gap seems to be on hiring and retaining niche cyber talent, as well as utilizing outsourcing strategies to remain agile and optimize operational processes[213].

The implications of this gap are profound: it not only leaves critical infrastructure and sensitive information vulnerable to attack but also implies higher costs for businesses and society as a whole due to the potential damages from cyber breaches. The economic impact is thus magnified by the rising cost of cybersecurity breaches and the challenges in securing a workforce capable of defending against an asymmetric warfare landscape, where attackers need only find one vulnerability, while defenders must secure against all possible threats.

1.2.3 The Role of AI in Bridging the Gap

This is where Artificial Intelligence (AI) steps in as a game-changer. AI's ability to learn, predict, and adapt can make

cybersecurity efforts more efficient and effective. AI systems can analyze vast datasets quickly and identify threats or abnormal patterns that a human might miss or take too long to identify. By automating threat detection and response, AI can perform repetitive tasks at a scale and speed unmatchable by human teams, thus freeing cybersecurity professionals to focus on more strategic defense aspects.

AI enhances threat detection and response efficiency by analyzing large datasets and identifying anomalies faster than humans. It automates repetitive tasks, allowing professionals to focus on strategic aspects. AI is crucial in real-time threat detection, advanced threat intelligence, and security task automation. It effectively combats phishing, ransomware, and zero-day attacks through predictive analytics. AI's behavioral analytics are also vital in detecting Advanced Persistent Threats (APTs) and insider threats by identifying unusual user behaviors.

Artificial Intelligence (AI) is becoming increasingly instrumental in cybersecurity, functioning as a force multiplier against cyber threats. Its predictive capabilities allow for a proactive defense, identifying potential dangers before they escalate. Real-time threat detection and response minimize the potential damage from cyberattacks. AI also excels in advanced threat intelligence by understanding criminal tactics and automating security tasks, thereby allowing human professionals to focus on complex strategies[3]. Moreover, AI's advanced cyber threat hunting proactively combats threats, and its future in cybersecurity looks to be one of expanded roles in detection, response, and prevention.

For instance, AI-powered security solutions can identify phishing attacks by analyzing the email content, sender details, and other attributes, and they do so with a speed and accuracy that traditional anti-phishing tools cannot match. In the case of ransomware attacks, AI can help back up data and systems efficiently and ensure quick recovery, reducing the downtime and potential damage from such attacks.

Moreover, AI's predictive capabilities are particularly potent against zero-day attacks. By analyzing data from previous breaches, AI can predict and identify vulnerabilities that might be exploited in the future, enabling preemptive action even before a zero-day exploit is attempted.

In combating APTs and insider threats, AI's behavioral analytics can play a pivotal role. By continuously learning and analyzing user behaviors, AI systems can detect anomalies that deviate from standard patterns, potentially indicating malicious activities.

1.2.4 The Economic and Socio-Political Implications of Cyber Threats

Cyberattacks can cause significant financial loss, damage customer trust, and harm businesses reputations, as seen in the Sony Pictures Entertainment hack. These attacks also pose national security risks by targeting critical infrastructure, exemplified by the Ukraine power grid attacks. Furthermore, cyber capabilities are increasingly used in geopolitical strategies, such as influencing elections, highlighting their role in international relations and democratic processes.

Beyond the immediate disruption caused by cybersecurity threats, there are far-reaching economic and socio-political consequences. Cyberattacks can undermine businesses' economic viability by causing direct financial loss, compromising customer trust, and tarnishing reputational equity. For instance, a ransomware attack can paralyze operations, leading to significant revenue loss, and the payment of the ransom often does not guarantee the restoration of data.

Moreover, sophisticated cyber-espionage campaigns against organizations can lead to the theft of intellectual property, loss of competitive advantage, and, in severe cases, the undermining of market position. The Sony Pictures Entertainment hack of 2014 is a case in point [61], where the theft and subsequent leak of company data had wide-ranging implications for the company's brand, financials, and strategic market position.

On a macro scale, consistent cyber threats against critical infrastructure pose national security risks. The attacks on Ukraine's power grid in 2015 and 2016[2], for instance, demonstrated how cyber warfare is assuming a dangerous dimension, with the potential to disrupt essential services and cause societal chaos.

2: Ukraine's electricity grid suffered cyberattacks believed to be orchestrated by Russian hackers, causing widespread power outages and raising concerns about critical infrastructure vulnerabilities.

There is also a geopolitical element at play. Nations are leveraging cyber capabilities not just for defensive purposes but for espionage, disruption, and influence [216]. The meddling in the 2016 United States elections is a stark reminder of how cyber capabilities can be weaponized to interfere in the democratic process, exacerbating political divisions and undermining citizens' trust in democratic institutions.

1.2.5 Proactive Cybersecurity Strategies

In light of the complex and evolving threat landscape, a reactive approach to cybersecurity is untenable. Organizations must adopt a proactive cybersecurity posture, one that involves staying ahead of threat actors. This involves continuous monitoring, threat intelligence, and the adoption of a 'zero trust' model[3], treating every attempt to access the orga-

3: The "zero trust" model is a cybersecurity approach that assumes no network or user can be trusted and requires continuous authentication and verification to protect against threats.

nization's system as potentially hazardous, even if it appears to come from within the organization's own network.

Emphasizing resilience is also crucial. It's unrealistic to assume that organizations can prevent every cyber-attack. Therefore, a robust cybersecurity strategy must include plans for quick detection, response, and recovery from breaches to minimize damage.

This section argues for a proactive, rather than reactive, approach to cybersecurity, emphasizing continuous monitoring, threat intelligence, and adopting a 'zero trust' model. Recognizing that not all attacks can be prevented, it stresses the importance of resilience and rapid response strategies. Given the cybersecurity skills gap, this poses a challenge, particularly for smaller organizations.

However, maintaining a proactive and dynamic cybersecurity posture is resource-intensive, requiring continuous investment in technology and human capital. Aligned with this goal, the world's big tech companies accumulated an investment of approximately $2.4 billion in 2021 as shown in Figure 1.3. With the current cybersecurity skills gap, this is a significant challenge for many organizations, particularly small and medium enterprises (SMEs) with limited resources.

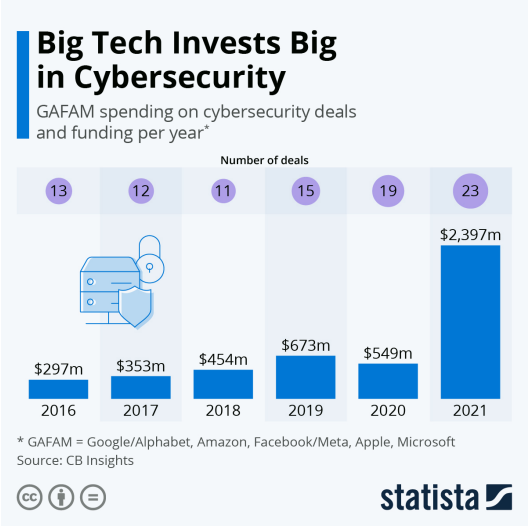


Figure 1.3: In 2021, GAFAM spent a combined $2.4 billion on funding or acquiring 23 cybersecurity companies, an increase of roughly $1.8 billion or 336 percent according to data aggregated by CB Insights. [288].

This resource challenge highlights the necessity of leveraging AI in cybersecurity strategies. AI's automation and predictive capabilities can compensate for the lack of human resources, enhancing an organization's ability to maintain a dynamic cybersecurity stance. By employing AI-driven security tools, organizations can efficiently detect and respond to threats in real time, predict future threat vectors, and adapt their security measures to evolving risks.

1.3 Artificial Intelligence and Its Current Applications to Cybersecurity

In the face of the escalating cyber threat landscape, Artificial Intelligence (AI) presents transformative potential in bolstering cybersecurity defenses. AI, in its essence, encompasses systems that can learn, reason, and act for themselves. They can process vast quantities of data and undertake complex computations, often far surpassing human capabilities. Within cybersecurity, AI technologies, particularly Machine Learning (ML) and Deep Learning (DL), are increasingly pivotal in predicting, identifying, and neutralizing cyber threats.

AI performs well in processing large volumes of data and outperforms human capabilities in threat detection. ML allows systems to learn from data patterns autonomously, while DL, an advanced form of ML, allows to independently identify and counter evolving cyber threats, including mutating malware.

Machine Learning, a subset of AI, involves systems that improve their performance without direct human intervention. They learn from patterns and insights derived from analyzing large data sets. Deep Learning, a further subset of ML, structures algorithms in layers to create an "artificial neural network"[4] that can learn and make decisions on its own. This capability is crucial in the detection of malware and malicious activity, even if the threats mutate or evolve.

4: An artificial neural network is a layered computational model that processes information using interconnected nodes to perform complex data analysis and pattern recognition.

1.3.1 Techniques of Artificial Intelligence

Artificial Intelligence (AI) is a broad field with a variety of techniques and subfields, each with its unique strengths and applicability to different problems in cybersecurity. Understanding these techniques is essential to appreciate how AI can be tailored to address specific cybersecurity challenges.

Machine Learning (ML): At its core, ML is about the construction of algorithms that enable computers to learn from and make decisions or predictions based on data. ML can be divided into three primary types:

- *Supervised Learning:* The algorithm is trained on a labeled dataset, meaning it's provided with explicit instructions on what to learn. An example would be training a model with labeled malware code so it can identify malware mutation[5].

5: Malware mutations refer to the dynamic alterations in the code and behavior of malicious software, aimed at evading detection and enhancing effectiveness.

- *Unsupervised Learning:* The algorithm must find patterns and relationships in datasets without any labels. This type is crucial for anomaly detection, a common practice in cybersecurity, as it sifts through data and identifies unusual patterns that may indicate a cyber threat.
- *Reinforcement Learning:* The algorithm learns by trial and error to achieve a clear objective. It makes decisions in a sequence of steps, learning from past actions and adjusting future ones to achieve its goal. This method could be used in cybersecurity simulations to find optimal strategies against cyber adversaries.

Deep Learning (DL): A subset of machine learning, DL structures algorithms in layers to create an "artificial neural network" that can learn and make intelligent decisions on its own. DL is particularly effective in processing unstructured data, which is prevalent in cybersecurity.

- *Convolutional Neural Networks (CNNs):* These are particularly suited for processing images, which can be applied in cybersecurity for tasks such as analyzing data visualizations or even for facial recognition in biometric security.
- *Recurrent Neural Networks (RNNs):* Effective for processing sequences of data, making them applicable in areas like time-series analysis for network traffic, which can help detect irregularities indicating potential cyber threats.

Natural Language Processing (NLP): This involves the ability of computers to understand and interpret human language. In cybersecurity, NLP can be used for automating the analysis of human-written content in emails or on the web to detect phishing attempts and understand the context in communication patterns.

Expert Systems: These are AI-based computer systems that emulate the decision-making ability of a human expert. In cybersecurity, expert systems can be used to make informed decisions based on a repository of knowledge about past cyber-attacks and known threats.

Graph Analytics: This technique uses graph structures for semantic queries with nodes, edges, and properties to represent and store data. Its ability to uncover relationships between

disparate data points makes it invaluable in cybersecurity for detecting hidden patterns and uncovering fraud rings.

Adversarial AI: In the cybersecurity context, this involves AI systems designed to combat other AI systems, used in scenarios like testing network defenses against AI-driven cyber-attacks or understanding how malware designed using AI might evolve over time.

1.3.2 Success Stories of AI in Cybersecurity

AI significantly enhances cybersecurity across various sectors by detecting anomalies, automating responses, predicting threats, identifying phishing, monitoring insider threats, and enriching threat intelligence, proving crucial in combating cyber attacks.

AI's impact on cybersecurity is not merely theoretical; its efficacy is demonstrated through its growing implementation across sectors and its role in defeating cyber attacks. Here are some notable applications:

Anomaly Detection: Traditional security systems that rely on signature-based approaches are often inept at detecting zero-day exploits or previously unseen malicious activity. AI-driven systems, with their ability to continuously learn and adapt from data, can identify and flag activities that deviate from the norm. For instance, Darktrace uses AI to spot abnormal behavior within networks in real time, allowing for the immediate neutralization of threats.

Automated Response to Incidents: Speed is crucial when mitigating cyber threats. AI systems can automate the response process, often containing the threat faster than a human could. Companies like Cylance[6] use AI to predict, prevent, and eliminate advanced threats, significantly reducing the time between threat detection and response.

6: Cylance develops anti-virus and other computer software that prevents viruses and malware.

Predictive Capabilities: AI can forecast future threats and security incidents by analyzing trends and patterns from various data sources. This predictive capability enables organizations to preemptively fortify their defenses, potentially stopping cyber threats before they materialize.

Phishing Detection: AI technologies can analyze emails for signs of phishing, often more accurately and rapidly than traditional methods. By examining both the content and context of communications, AI can identify discrepancies that may indicate phishing attempts, even if they are from a known contact.

Insider Threat Detection: AI can help monitor user behaviors, detect unusual patterns, and alert organizations to

potential insider threats. For example, DTEX Systems[7] combines analytics with AI to detect insider threats, prevent data breaches, and secure intellectual property by monitoring user behavior.

7: DTEX Systems works integrating advanced analytics and AI to continuously monitor user behavior within an organization. This involves analyzing patterns of user activities and identifying deviations from normal behavior[74].

Enhancing Threat Intelligence: AI can analyze vast datasets from diverse sources in real time, delivering enriched threat intelligence. It can identify emerging vulnerabilities, malware, and other cyber threats, providing security teams with actionable insights that can inform their security strategies.

1.3.3 Challenges and Ethical Considerations in AI-Enabled Cybersecurity

Despite its considerable benefits, the integration of AI into cybersecurity is not without its challenges. One significant concern is the potential for AI systems to be biased, which can occur if the data they are trained on is not fully representative. This bias can lead to false positives or negatives in threat detection, undermining the reliability of these systems.

Additionally, as AI systems become more integrated into cybersecurity, there's a growing risk of threat actors using AI for malicious purposes. For example, adversaries could potentially use AI to automate cyber attacks, conduct extensive reconnaissance activities, or even create more sophisticated malware that can learn and adapt to security measures. This potential arms race between cybercriminals and security professionals is a looming challenge.

While AI offers significant advantages in cybersecurity, it faces challenges such as the potential bias of unrepresentative training data, leading to inaccurate threat detection. In addition, there is the risk of cybercriminals using AI for malicious purposes, creating a potential arms race. Ethical issues also arise around privacy and surveillance, requiring a balance between security and privacy rights. Strategic implementation and continued research are essential to maximise the ethical and effective use of AI in cybersecurity.

Ethically, the use of AI in cybersecurity raises questions about privacy and surveillance [171]. The extensive monitoring capabilities of AI could lead to infringements on privacy if not managed judiciously. Organizations will need to balance the enhanced security provided by AI with individuals' rights to privacy, necessitating a framework of best practices and regulations.

The application of AI in cybersecurity offers promising avenues to combat the increasingly sophisticated cyber threat landscape effectively. However, its implementation must be strategic, with considerations for its challenges and ethical implications. As AI continues to evolve, ongoing research, development, and dialogue will be crucial in maximizing

its potential in a manner that is ethical, effective, and equitable.

1.4 Hypothesis and Objectives

Within the dynamic landscape of cybersecurity, this research pivots on the following fundamental hypothesis that underline the potential transformation AI can bring into this domain:

My research question investigates how artificial intelligence in cybersecurity improves threat detection and response over traditional systems, focusing on real-time data processing, adaptability to emerging threats, and proactive attack prediction and neutralization.

"The application of artificial intelligence in cybersecurity significantly enhances threat detection and response, enhancing traditional systems due to its real-time data processing, adaptability to emerging threats, and capability to proactively predict and neutralize attacks."

This hypothesis underlines the premise that AI, through its advanced computational and learning abilities, can outperform traditional cybersecurity systems in both speed and efficiency, providing a more robust defense against the cyber threats.

This thesis is grounded on a central objective and several specific objectives, designed to guide the research in a structured and comprehensive way.

Main Objective: To explore and analyze the impact and implementation of artificial intelligence within the realm of key cybersecurity domains, including malware detection, network traffic analysis, spam filtering, and adult content identification. This comprehensive assessment aims to delineate AI's strengths, limitations, and the opportunities it presents for the advancement of cybersecurity measures. Through qualitative and quantitative analyses, the research seeks to provide actionable insights and contribute to the development of more robust and effective AI-driven cybersecurity strategies.

Expanding on the main objective, the following specific objectives explore different areas of the intersection between AI and cybersecurity, each of which sheds light on different dimensions of the research.

Literature Study: Conduct a comprehensive literature review to understand the current state of cybersecurity and how artificial intelligence has been integrated into this field so far.

Evaluation of AI Applications in Cybersecurity: Critically assess how AI technologies have been implemented to tackle challenges in malware detection, network anomaly detection and spam identification. This evaluation will highlight areas where AI has demonstrated significant advantages over traditional methods in key cybersecurity domains.

Own Research: Present and discuss the research carried out to tackle four critical and demanding areas within cybersecurity, providing empirical evidence of the associated benefits and challenges. Each area has been selected due to its significant impact on the overall security posture of organizations and the potential benefits AI can bring to these domains.

- Malware Detection: Detail the development and testing of AI models aimed at identifying and classifying malware, showcasing the effectiveness of these models compared to conventional approaches.
- Network Traffic Analysis: Describe the application of AI in analyzing network traffic to detect suspicious activities, emphasizing the model's accuracy and efficiency.
- Spam Filtering: Share insights from employing AI techniques for spam detection, focusing on the reduction of false positives and the improvement of filtering accuracy.
- Adult Content Filtering: Examine the use of AI in identifying and filtering adult content, highlighting how these systems can be implemented to protect users and ensure compliance with digital content standards.

Anticipation of Future Challenges: Based on research and available data, anticipate how cyberattacks will evolve in the future and how artificial intelligence can prepare us to face these emerging challenges.

Strategic Recommendations for AI Integration : Provide actionable recommendations for organizations looking to integrate AI into their cybersecurity frameworks, based on the research outcomes. These recommendations will

reflect both the potential of AI to enhance cybersecurity defenses and the considerations necessary to mitigate associated risks.

1.5 Research Methodology

To answer the research question, the thesis uses a robust, cyclic research methodology to explore AI in cybersecurity, generating insights and evolving with feedback from the scientific community and society. It combines various approaches to provide insights and practical solutions to current cybersecurity challenges.

This thesis is grounded in a robust and multifaceted research methodology, designed to thoroughly explore the intersection of artificial intelligence and cybersecurity. Through a series of complementary approaches, this research aims to generate valuable insights and practical solutions to contemporary challenges in the field of cybersecurity. Furthermore, this methodology is cyclic in nature, feeding on the impact and feedback received from both the scientific community and society at large, and in turn, enriching the field with the resultant work.

Literature Review: The first phase of the research involves a systematic review of the existing literature. This stage consolidates a foundational understanding of current themes in cybersecurity and artificial intelligence, identifying gaps in the existing literature, and setting the context for subsequent empirical studies.

Empirical Studies Design: Following the literature review, empirical studies focused on specific problems within cybersecurity will be designed and conducted. Each study will be formulated to investigate how artificial intelligence can address these issues innovatively and effectively.

Quantitative Evaluation: Quantitative methods will be used to measure the performance and effectiveness of the proposed AI solutions. This may include, but is not limited to, threat detection accuracy, response speed, and resource efficiency.

Validation with the Scientific Community: The findings and methodologies of the research will be presented to the scientific community for scrutiny and validation. This will include presentations at conferences, publications in journals, and the use of collaborative platforms for academic exchange.

Adaptation with Received Feedback: Feedback received from the scientific community will be instrumental in refin-

ing and enhancing the proposed approaches and solutions. This iterative process strengthens the research and ensures that it remains aligned with academic standards and practical needs.

Validation with Real-World Samples: To ensure real-world applicability, the developed AI solutions will be tested and validated using real datasets and environments. This critical phase aims to confirm the feasibility and effectiveness of the solutions in practical scenarios and under authentic operational conditions.

Knowledge Transfer: Finally, knowledge dissemination is a key stage. The research findings will be broadly shared with society, not just through academic publications and presentations at conferences, but also through outreach initiatives and educational platforms to foster awareness and education in cybersecurity.

By weaving together these components, the methodology of this thesis seeks not only to advance academic knowledge in the field of cybersecurity and AI but also to incite practical and sustainable changes in how cybersecurity is practiced and understood in the broader society.

1.6 Document Structure

The doctoral thesis presented in this document is organised in the following chapters: (i) Introduction, (ii) Literature Review, (iii) Malware Detection: Neural Insights, (iv) NetFlow Defense: CNN Surveillance, (v) PE Malware Analysis: DNN Exploration, (vi) Content Filtering: Adult Imagery, (vii) Spam Analysis: LSTM Application, and (vii) Conclusions.

Chapter 1 explores the landscape of cybersecurity, starting with an examination of the motivation behind the study. It discusses today's cybersecurity challenges, including the types of cyber threats, the skills gap and the role of AI in addressing these issues. The chapter also discusses the economic and socio-political implications of cyber threats and proactive cybersecurity strategies. Furthermore, it provides insights into how artificial intelligence is currently applied in cybersecurity, covering techniques, success stories, and ethical considerations. The chapter concludes by outlining the research's hypothesis, objectives, and the methodology to be employed in the study.

Chapter 2 takes an in-depth look at two pivotal domains: cybersecurity and artificial intelligence (AI). Within this chapter, an overview of cyber security is provided, including its history, contemporary methods and limitations. Simultaneously, it navigates through the world of AI, tracing its evolution, examining its current state, understanding its various types, and highlighting the challenges it confronts. Furthermore, the chapter underscores the intersection of these two realms by investigating the application of AI in cybersecurity, assessing its impact, acknowledging its limitations, and illuminating potential future research directions.

Chapter 3 focuses on malware detection and provides information on the growing threat of mobile malware. It discusses the evolution of malware and countermeasures. Deep learning's role in malware detection is explored, with a focus on a neural approach for classification using Android bytecode. The chapter presents research that has been carried out on a methodology for detecting malware on android by training convolutional neural networks for image-based detection. It presents experimental results, performance metrics, and concludes with pathways for future research.

Chapter 4 explores network security and introduces a NetFlow defense system using CNN surveillance. It begins with an overview of network security, highlighting its importance and evolution. The chapter looks in more detail at the challenges, including types of cyber-attacks, emerging threats, vulnerabilities and human factors.. It emphasizes the need for innovative approaches and discusses advances in network security. Shows a methodological exploration carried out in a research study on the detection of malicious network traffic by analysing network packets. It also presents an in-depth analysis of the findings and results, as well as future lines of research.

Chapter 5 explores the analysis of Portable Executable (PE) malware using deep neural networks (DNN). It begins with the fundamentals of PE files and their current state in software and malware. The historical evolution of malware in PE files, from early instances to modern trends, is explored. The chapter also covers the use of DNNs in malware detection, assessment criteria for DNN methods, and the implementation of DNNs for PE malware detection, including methodology, results, and analysis.

Chapter 6 focuses on content filtering, particularly addressing adult imagery. It begins with a survey of the adult content landscape, emphasizing the dangers of deepfakes. The chapter discusses ethical challenges in combating explicit content and the role of AI in this context. It outlines the experimental methodology for content filtering carried out in a experiment by explaining data gathering, method insights, and outcomes.

Chapter 7 explores the topic of spam analysis with a focus on the application of Long Short-Term Memory (LSTM). It begins by discussing the phenomenon of spam and the landscape of spam. The chapter traces the evolution of spam detection methodologies and elaborates a research on the application of LSTM in spam detection, including methodology and results. It concludes by synthesizing insights on how LSTM contributes to advancing spam detection.

Chapter 8 provides the conclusion of the research. It summarizes the research objectives and highlights contributions to the literature, comparing them with previous work and emphasizing originality. The chapter presents research findings, discusses limitations, and suggests future research directions. It concludes with overall conclusions, their relevance, and final considerations, including lessons learned and personal growth as a researcher.

Literature Review 2

2.1 Cybersecurity Landscape

In the digital era, where the reliance on computer systems, networks, and data is ever-increasing, the significance of cybersecurity cannot be overstated. As a multifaceted and evolving field, cybersecurity is fundamental in protecting sensitive information and maintaining the integrity of technology-driven systems in both personal and professional realms. This section aims to elucidate the concept of cybersecurity, providing a comprehensive understanding of its various components and their importance in today's technologically driven world.

Cybersecurity is essential in the digital era, aiming to protect technology systems and sensitive data from threats through strategies, risk management, user education, compliance with regulations, safeguarding infrastructure, incident response, and continuous improvement. It's vital for personal and professional security.

Cybersecurity, at its core, is a set of strategies, practices, and technologies aimed at safeguarding computers, networks, applications, and data from attack, damage, or unauthorized access. This definition encapsulates a variety of aspects, including, but not limited to, the following:

- Protection of Digital Assets: Cybersecurity involves safeguarding digital assets, which include hardware, software, and data. The aim is to protect these assets from cyber threats such as viruses, malware, ransomware, and other forms of malicious software.
- Risk Management: It involves identifying, analyzing, and mitigating risks to the security of information systems. This process includes regular assessments of potential vulnerabilities within systems and the implementation of strategies to manage these risks effectively.
- User Education and Awareness: An often-underestimated aspect of cybersecurity is the role of user behavior. Educating users about safe practices, such as strong password policies, awareness of phishing attempts, and the secure handling of sensitive information, is crucial.
- Policies and Regulations Compliance: Cybersecurity also encompasses adherence to legal and regulatory requirements. Organizations must comply with various laws and regulations that govern data protection and

privacy, such as GDPR, HIPAA, or CCPA, depending on their geographical location and the nature of their business[1].

- Technological Infrastructure Safeguarding: This involves the deployment of physical and software-based tools to protect networks and systems. Firewalls, antivirus software, intrusion detection systems, encryption, and secure network architectures are examples of these tools.
- Incident Response and Recovery: Cybersecurity also deals with the ability to respond to and recover from security incidents. This includes having plans and procedures in place for incident response, disaster recovery, and business continuity in the event of a security breach.
- Continuous Monitoring and Improvement: As cyber threats evolve, so must cybersecurity measures. Continuous monitoring of systems and regular updates to security practices and technologies are necessary to stay ahead of potential threats.

1: Cybersecurity policies and regulations compliance is complex due to variations across countries, as each nation has its own unique standards and requirements.

The increasing reliance on digital technology across various sectors ranging from finance and healthcare to education and government has elevated cybersecurity to a matter of utmost importance. The financial ramifications of cyber attacks are profound, with potential losses stemming from the theft of corporate and financial information, disruption in trading, and the costs associated with system and network repairs. These financial impacts extend beyond individual organizations, touching upon the economic security of entire nations. Cybersecurity is integral to protecting the financial sector against attacks that could destabilize entire economies.

Cybersecurity is critical in protecting financial sectors, personal data, national security, and business operations against cyber threats, ensuring economic stability, privacy, public trust, and continuity. The evolving nature of threats with new technologies demands adaptive strategies, highlighting cybersecurity's global significance.

Moreover, the vast quantities of personal data stored and processed online necessitate robust cybersecurity measures to prevent unauthorized access and data breaches that could lead to identity theft and fraud. This protection of sensitive information is not just a matter of individual privacy but also a cornerstone of public trust.

In the realm of national security, cybersecurity acquires a heightened significance. Protecting critical infrastructure, including power grids, water supplies, and communication networks, is paramount[12, 53]. The potential for catastrophic consequences on public safety and national security due to cyber attacks on these systems cannot be overstated.

For businesses, cybersecurity is pivotal in ensuring operational continuity and integrity. Cyber attacks can disrupt business systems, leading to downtime and loss of productivity. A strong cybersecurity framework is essential for businesses to recover swiftly from such attacks and maintain their operational integrity. Additionally, adherence to various cybersecurity regulations and laws is crucial for businesses to avoid legal fines, making cybersecurity a legal imperative as well.

A company's reputation is closely linked to its ability to protect its data and systems against cyber threats. A single security breach can significantly damage a company's reputation, resulting in the loss of customers and partners. This aspect of cyber security underlines the importance of maintaining public trust.

The nature and complexity of cyber threats are constantly evolving. The emergence of new technologies such as the Internet of Things (IoT), cloud computing, and artificial intelligence (AI) has expanded the attack surface, necessitating increasingly sophisticated cybersecurity strategies[70].

2.1.1 History and Evolution of Cybersecurity

The journey of cybersecurity begins in the early 1970s with the emergence of the first computer virus, named "Creeper"[165]. This rudimentary form of a computer virus was more of an experiment than a malicious threat, displaying a simple message: "I'm the creeper, catch me if you can!" This marked the inception of what would become a long battle against various forms of computer threats.

As computer networks grew in size and complexity, so did the threats. A significant milestone was the Morris Worm of 1988[199], one of the first worms to spread across the Internet. Created by Robert Tappan Morris, it was intended to measure the size of the internet but ended up causing widespread disruption due to its replicative nature. This incident was a wake-up call, leading to the formation of the Computer Emergency Response Team (CERT)[2] and setting the stage for the development of cybersecurity protocols.

Concurrently, there was a rise in antivirus software, a direct response to the increasing prevalence of computer viruses.

The evolution of cybersecurity, starting in the 1970s, reflects a continuous battle against growing threats. The Morris Worm of 1988 prompted the creation of CERT and cybersecurity protocols. Antivirus software and firewalls emerged as defenses. Threats evolved from simple viruses to complex malware, ransomware, and state-sponsored attacks. Security measures advanced with encryption, IDS/IPS, and advanced authentication methods. The ongoing struggle between attackers and defenders demands constant innovation in cybersecurity strategies.

2: A Computer Emergency Response Team (CERT) is a dedicated group that monitors, responds to, and mitigates cybersecurity incidents and threats.

Companies like McAfee[175] and Gen Digital[3] became household names, offering solutions to protect personal and enterprise computers. Another major development was the advent of firewalls. Initially rudimentary packet filters, firewalls evolved into sophisticated systems capable of deep packet inspection and intrusion prevention, forming the first line of defense in network security.

3: Generation Digital encompasses a diverse range of products, including antivirus solutions like Avast, Norton, AVG, etc[69].

Over the years, the nature of cybersecurity threats has undergone a dramatic transformation. Initially, the world saw simple viruses, primarily causing inconvenience rather than serious damage. However, with the proliferation of the internet and digitalization, the scope of threats expanded. The 21st century witnessed the rise of complex malware, capable of stealing sensitive data and causing significant financial and reputational damage to individuals and organizations.

Furthermore, ransomware emerged as a formidable threat, encrypting victims data and demanding ransom for its release. Examples like WannaCry[50] and NotPetya[156] demonstrated not only the financial impact but also the potential for widespread disruption to critical infrastructure.

The landscape further complicated with the advent of state-sponsored cyber attacks. These attacks, often sophisticated and well-funded, target critical national infrastructure, steal intellectual property, and even attempt to influence electoral processes[92], as seen in various allegations of election interference.

4: Encryption plays a pivotal role in safeguarding data, with numerous types like AES, RSA, and more, enhancing security in diverse applications.

In response to these evolving threats, cybersecurity measures have also advanced significantly. Encryption became a standard practice[4], not only in securing communication channels but also in protecting data at rest. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) were developed to monitor network traffic for suspicious activity and take proactive measures to prevent breaches.

Moreover, the concept of advanced authentication methods, including biometrics and multi-factor authentication, gained traction[197, 235, 170]. These methods offered additional layers of security, moving beyond the traditional username and password, to guard against the increasing sophistication of phishing and other social engineering attacks.

2.1.2 Threats and Vulnerabilities

In the cybersecurity landscape, understanding threats and vulnerabilities is paramount as they represent the potential risks and weak points that cyber attackers exploit to compromise systems, steal data, and disrupt services in our increasingly digital world. In this section we take a closer look at critical aspects of cyber security, focusing on web application security and operating system threats. This section highlights the significance of protecting against vulnerabilities like XSS and SQL Injection, which pose substantial risks. Additionally, we cover the importance of securing operating systems against unauthorized scans, exploitation of vulnerabilities, and the deployment of rootkits and malware, underlining the necessity of robust cybersecurity measures in today's digital landscape.

Web Application Security

In the ever-evolving landscape of technology, Web Application Security stands as a critical pillar in safeguarding online data and services. As the reliance on web applications continues to escalate in various sectors including finance, healthcare, and e-commerce, it's imperative that these applications are not only functional but also secure from potential threats. This section dives into the intricate world of web application security, addresses the main challenges and offers ideas on effective security strategies.

The vulnerability known as XSS (Cross-Site Scripting) has historically been one of the most prominent in the cybersecurity landscape, but its prevalence has significantly decreased thanks to the growing awareness and dedicated efforts to prevent and mitigate it. Cross-Site Scripting is a prevalent security vulnerability in web applications, where attackers inject malicious scripts into otherwise benign and trusted websites. There were 3 primary types of XSS attacks[5], reflected, reflected and DOM-based XSS[142].

Example of XSS Attack

Let's consider a login page that displays an error message when incorrect credentials are entered, and this error message includes the username parameter. In this scenario,

5: Reflected XSS occurs with direct user input reflection in responses, Stored XSS through server-side saved user inputs, and DOM-Based XSS by altering the DOM in the browser, each exploiting unsanitized data in different ways within web applications.

we can input the following sentence as the username: "<script>alert()</script>". When the error response is generated and displayed, it triggers an alert on the website.

For many years, Stored, Reflected, and DOM XSS were perceived as distinct categories of Cross-Site Scripting attacks. However, it's become evident that these categories are not mutually exclusive and often overlap. For instance, XSS attacks can be both Stored and Reflected within the context of DOM-Based or Non-DOM-Based XSS. This can lead to some confusion. To bring clarity, around 2012, the research community introduced two new terms to better categorize XSS incidents Server XSS and Client XSS[6].

6: Server XSS involves server-side inclusion of untrusted user data in responses, leading to potential Reflected and Stored XSS vulnerabilities, while Client XSS occurs through unsafe JavaScript calls modifying the DOM with untrusted data, encompassing both Reflected and Stored XSS types[273].

Considering that both Server XSS and Client XSS can manifest as either Stored or Reflected, this terminology yields a 2x2 matrix. It delineates Client & Server XSS on one axis and Stored and Reflected XSS on the other axis, as illustrated in Dave Witchers' DOM Based XSS presentation (Figure 2.1).

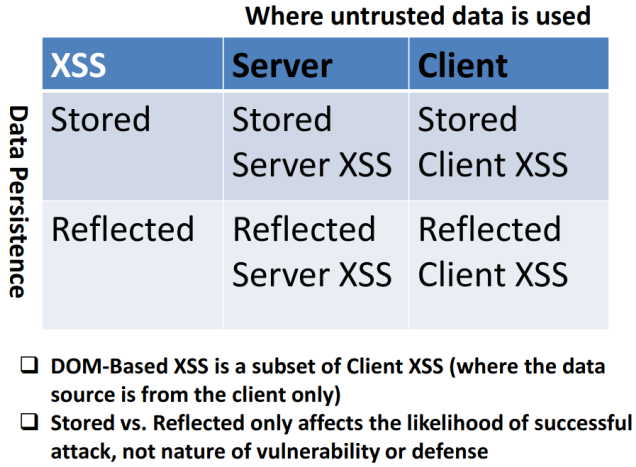


Figure 2.1: A 2x2 Matrix Mapping Client & Server XSS vs. Stored and Reflected XSS [273].

Another of the most well-known vulnerabilities alongside XSS is SQL Injection and it remains one of the most hazardous threats to web applications. This attack involve the manipulation of a SQL query through client-provided data input into an application. Successful exploitation of this vulnerability can lead to various malicious outcomes, such as reading sensitive information from the database, altering data (including Insert, Update, or Delete operations), performing administrative tasks on the database (like shutting down the DBMS), accessing specific files on the DBMS's file system, and in some scenarios, executing commands on the operating system. These attacks fall under the broader

category of injection attacks, where SQL commands are inserted into data-plane inputs to manipulate the execution of pre-established SQL commands.

Example of SQL Injection Attack

Imagine a website with a login form that checks user credentials against a database. The backend code might look something like this in a vulnerable system "`SELECT * FROM users WHERE username = \$username AND password = \$password;`". In this code, *username* and *password* are variables that take user input from the login form. An attacker can exploit this by entering SQL code into the input fields. For example, if the attacker inputs is "admin' –" the SQL query becomes `SELECT * FROM users WHERE username = 'admin'` because the rest is commented by scaping the sentence with a single quote.

OS Threats

Operating system (OS) security is a pilar of overall system integrity and confidentiality. Within this scope, threats such as unauthorised scanning of ports and services, exploitation of vulnerabilities and deployment of rootkits and malware present significant risks. This section provides an in-depth look at these threats, providing a technical review and practical examples.

Port and service scans are early steps in the reconnaissance phase of a cyber-attack. Tools such as Nmap[169] and Nessus[251] are frequently used to discover open ports and detect services running on a target system. To mitigate these threats, firewalls and traffic filtering rules must be meticulously configured to block unauthorized access and obscure system details from external scans. Implementing Intrusion Detection Systems (IDS) can also help by alerting on suspicious scanning activities.

Example of NMAP Tool

For example, an Nmap command "`nmap -sV target_ip`" can identify open ports and service versions, potentially revealing vulnerable services.

Vulnerability Exploitation is another fundamental part of OS vulnerabilities. Exploiting vulnerabilities in OS components or services can grant attackers unauthorized access or control. Zero-Day vulnerabilities, which are previously unknown flaws, pose a particularly acute threat due to the absence of available patches. Common Vulnerabilities and Exposures (CVEs[7]) database provides a reference for known vulnerabilities, aiding in risk assessment and mitigation strategies. For instance, CVE-2019-0708[192], also known as "BlueKeep," is a critical vulnerability in Microsoft's Remote Desktop Protocol (RDP) that allows for remote code execution. Effective patch management is crucial; it involves regularly updating systems and applying security patches to mitigate known vulnerabilities.

7: CVEs, or Common Vulnerabilities and Exposures, are standardized identifiers used to track and reference known security vulnerabilities in software and hardware.

Rootkits, on the other hand, are sophisticated types of malware designed to hide their presence and other malicious activities, making them difficult to detect. They often replace or modify core OS components. An example is the infamous "Stuxnet"[24] worm, which targeted SCADA systems[8] and used rootkit techniques to conceal itself. Malware, on the other hand, encompasses a broad range of malicious software, including viruses, worms, and spyware. Detection techniques involve signature-based methods, where antivirus software compares files against a database of known malware signatures, and heuristic-based methods, which analyze behaviors for suspicious patterns. Forensic analysis plays a pivotal role in post-incident investigations, using tools like Volatility for memory analysis and Autopsy for comprehensive digital forensics, to uncover how the malware operated and the extent of the compromise.

8: SCADA, or Supervisory Control and Data Acquisition, systems are industrial control systems that monitor and control critical infrastructure and industrial processes.

Malware & Binary Analysis

In the field of cybersecurity, malware analysis is a critical discipline that involves dissecting malicious software to understand its functionality, origin, and potential impact. This process employs various techniques, with reverse engineering and sandboxing being paramount for a detailed examination of malware behavior.

Reverse engineering is a fundamental technique in malware analysis, allowing analysts to deconstruct malware and explore its inner workings. This process typically involves the use of disassemblers and debuggers to translate binary code

into a more human-readable form. Disassemblers like IDA Pro or Ghidra break down the executable code into assembly language, providing insights into the malware's operational logic. Debuggers, such as x64dbg or OllyDbg, allow analysts to execute the malware in a controlled environment, stepping through code execution to observe behavior and effects in real-time.

Malware analysis in cybersecurity involves dissecting malicious software to understand its functionality and impact. Reverse engineering is essential, using disassemblers to translate binary code and debuggers to observe behavior. Static analysis examines code structure, while dynamic analysis observes real-time behavior. Sandboxing provides a safe environment to analyze malware execution patterns.

A practical example of reverse engineering can be seen in the analysis of a Trojan malware. By disassembling the binary, an analyst might uncover the sequence of instructions that enables the Trojan to open a backdoor for remote access. Further, using a debugger to step through the code execution might reveal the malware's network communication routines, exposing command and control (C2) server[9] addresses or data exfiltration mechanisms.

9: A Command and Control (C2) server is a centralized system that manages and communicates with compromised devices.

Static Analysis forms part of the reverse engineering process, where the malware is examined without executing the code. This approach focuses on reviewing the code structure, strings, API calls, and other static properties to infer the malware's capabilities and intentions. Static analysis can quickly highlight suspicious or malicious indicators within the code, such as obfuscated strings or known malicious function calls.

Dynamic Analysis, on the other hand, involves executing the malware in a controlled environment to observe its behavior and interaction with system resources in real-time. This method is invaluable for understanding the malware's runtime operations, such as registry modifications, file creation, and network communications.

Sandboxing complements reverse engineering by providing an isolated environment to safely execute and analyze malware without risking the host system or network. Tools like Cuckoo[10] Sandbox automate the execution of malware samples and capture their behaviors, including system calls, network traffic, and changes to the filesystem. There are tools that offer sandboxing functionalities, such as seccomp in linux, which controls system calls by applications (Figure 2.2). Sandboxing enables the capture of malware's behavior patterns and its interaction with external systems, which might not be evident through static analysis alone.

10: Cuckoo Sandbox is essentially an open-source or free software that automates malware analysis on Windows, Linux, macOS, and Android devices.

For instance, when analyzing a piece of ransomware, sandboxing can reveal its encryption routines, file targeting patterns,

and ransom note creation, without actually compromising real data. The sandbox environment can simulate user interactions and system responses, allowing for a comprehensive view of the malware's execution flow and side effects.

Penetration Tools & Testing

In the domain of cybersecurity, Penetration Tools and Testing play an instrumental role in identifying and mitigating vulnerabilities within software and network systems. This section takes a closer look at the technical aspects of Static & Dynamic Code Scanning and Penetration Testing, providing a practical overview of their application.

Static & Dynamic Code Scanning are foundational components of a secure software development lifecycle[64]. Static Application Security Testing (SAST) tools, such as SonarQube or Checkmarx, analyze source code at rest to detect security vulnerabilities without executing the code. These tools scrutinize code syntax and structure to identify issues like SQL injection or cross-site scripting (XSS). For instance, a SAST tool might flag a piece of code that concatenates user input directly into a SQL query, indicating a potential SQL injection vulnerability.

Dynamic Application Security Testing (DAST) tools, on the other hand, analyze running applications to identify vulnerabilities that manifest during execution. Tools like OWASP ZAP or Burp Suite act as a client to interact with the application, testing it from the outside to identify runtime issues such as authentication problems, exposed sensitive data, or session management weaknesses. An example of DAST in action could involve automated fuzz testing[11] against a web application's inputs to uncover unhandled exceptions or errors that could lead to vulnerabilities.

11: Fuzz testing is a technique involving automated input of random, invalid, or unexpected data to detect vulnerabilities and software bugs.

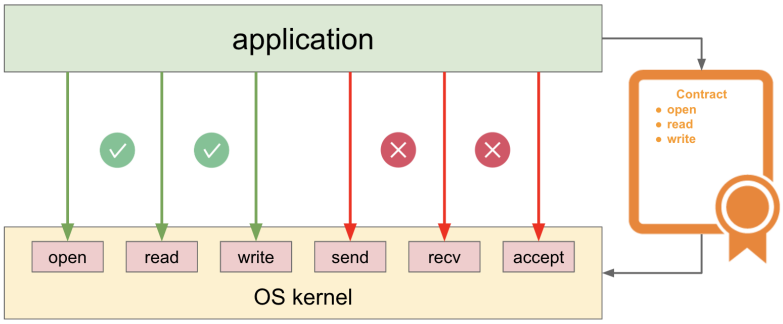


Figure 2.2: Linux seccomp allows applications to declare intended system call use to the kernel, enhancing security by restricting unnecessary services, like preventing a file conversion app from accessing network services it doesn't need[58].

Post-testing remediation involves addressing identified vulnerabilities, followed by re-testing to ensure fixes are effective. This cycle is critical for maintaining the security integrity of the application.

Penetration Testing simulates cyber-attacks against your computer system to check for exploitable vulnerabilities. In the context of web application security, penetration testers use methodologies like the OWASP Testing Guide[12] to systematically identify and exploit security weaknesses.

Penetration testing tools range from network scanning tools such as Nmap, which we discussed in the previous section, to exploit frameworks such as Metasploit[210], which provides a vast repository of exploits for known vulnerabilities. A practical example of penetration testing might involve using Nmap to discover a vulnerable service running on a server, then leveraging Metasploit to exploit this service using a known vulnerability, such as EternalBlue[135] for SMB[13], to gain unauthorized access.

12: The OWASP Testing Guide is a comprehensive resource providing guidance and best practices for testing web applications and software security.

13: SMB (Server Message Block) is a network protocol used for file sharing, printer access, and communication between devices in a local network.

Regulations & Standards

In the realm of cybersecurity, adherence to Regulations & Standards is not just a legal obligation but a cornerstone of trust and reliability in the digital age. This section discusses the background to GDPR and ISO 27001, highlighting its requirements, implementation strategies and the importance of compliance.

The General Data Protection Regulation (GDPR) sets a global standard for data protection and privacy, imposing strict data processing practices on organisations operating in the European Union or targeting individuals. Key requirements include lawful processing of personal data, explicit consent for data collection, and the right to data portability and erasure. Non-compliance can result in penalties up to 4% of annual global turnover of the company or €20 million, whichever is higher (Figure 2.3). A practical implementation step is the appointment of a Data Protection Officer (DPO) responsible for overseeing data protection strategies and ensuring compliance.

ISO 27001 is a globally recognized framework for managing information security, focusing on the establishment, implementation, maintenance, and continuous improvement of an

Information Security Management System (ISMS). Certification involves a two-stage audit process by an accredited certification body. The first stage assesses the ISMS documentation, and the second stage evaluates the effectiveness of the ISMS in practice. For example, an organization might implement access control measures, ensuring that employees have access only to the information necessary for their role, as part of their ISMS.

Compliance with these standards involves a comprehensive approach that includes policy development, employee training, regular audits, and the adoption of technical measures such as encryption and intrusion detection systems. Organizations often employ tools like compliance management software to streamline the process, ensuring that all regulatory requirements are met and maintained continuously.

2.1.3 Case Studies and Analysis of Methods

In recent years, several high-profile cyber attacks have reshaped our understanding of cybersecurity threats and responses. Three significant cases stand out: the WannaCry ransomware attack, the Sony Pictures hack, and the SolarWinds breach.

Recent high-profile cyber attacks like WannaCry, Sony Pictures hack, and SolarWinds breach underscore the critical need for robust cybersecurity, emphasizing software updates, insider threat awareness, and supply chain security to mitigate and prevent future threats.

WannaCry Ransomware Attack: In May 2017, the WannaCry ransomware spread globally, affecting more than 200,000 computers across 150 countries[184]. This attack exploited vulnerabilities in older Windows operating systems (Figure 2.4), encrypting data and demanding ransom for its release. Critical infrastructures, including hospitals and transportation systems, were severely impacted. WannaCry highlighted the importance of regular software updates and the dangers of using unsupported operating systems.

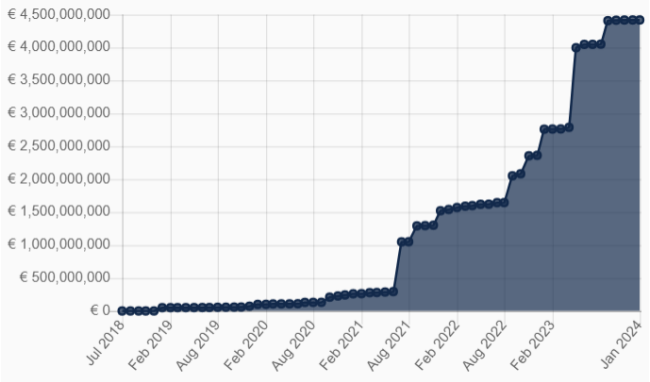


Figure 2.3: Graph shows how many fines and what sum of fines have been imposed per month since 2018. It t overview contains a cumulative summary, that is, all fines accumulated up to each month. [255].

Sony Pictures Hack: In 2014, Sony Pictures Entertainment became the target of a devastating cyber attack. Hackers, allegedly backed by North Korea, gained access to the company's network, stole sensitive data, and caused substantial financial and reputational damage[61]. This incident underscored the risks of geopolitical cyber warfare and emphasized the need for robust data protection and network security measures.

SolarWinds Breach: The SolarWinds breach, discovered in 2020, was a sophisticated supply chain attack. Malicious actors compromised the software of SolarWinds, a company that provides network management tools, and used this access to infiltrate the networks of numerous organizations, including U.S. government agencies. This breach highlighted the vulnerabilities in supply chain security and the complexity of defending against state-sponsored cyber espionage[7].

In response to these attacks, various cybersecurity methods were employed.

WannaCry Response: The immediate response to WannaCry involved the deployment of patches for the exploited vulnerabilities and the isolation of affected systems. The attack's spread was inadvertently halted by a security researcher who discovered and activated a "kill switch" in the malware's code[14]. Post-incident, there was a renewed emphasis on cybersecurity awareness, regular software updates, and the decommissioning of outdated systems.

14: A "kill switch" in malware code is a security mechanism designed to deactivate the malicious software remotely or under specific conditions.

Sony Pictures Response: The response to the Sony Pictures hack involved a combination of forensic analysis to understand the breach's extent and the implementation of enhanced

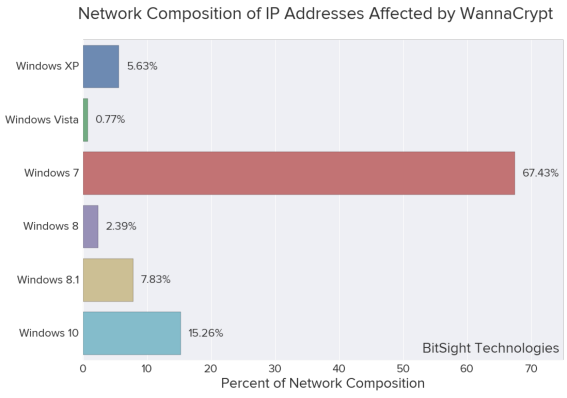


Figure 2.4: This chart shows the operating system distribution for IP addresses linked to WannaCry infections in 2017, not necessarily individual infected machines. Some IPs may represent multiple devices, not all of which were affected[32].

security protocols. The incident led to a greater focus on insider threat detection, employee training on cybersecurity practices, and the importance of strong perimeter defenses.

SolarWinds Response: Addressing the SolarWinds breach required a coordinated effort involving software updates to remove the backdoors[15], thorough network analysis to detect any presence of the attackers, and a reevaluation of network security, particularly concerning third-party vendors. It underscored the need for rigorous software testing, continuous monitoring, and a more robust approach to supply chain security.

15: Backdoors are covert access points embedded in software, allowing unauthorized users or processes to bypass normal authentication and gain control.

The effectiveness of cybersecurity strategies in these cases varies. While reactive measures were essential in mitigating immediate damage, these incidents highlight the necessity of proactive strategies. Regular software updates, comprehensive security audits, employee training, and a deeper understanding of the entire digital supply chain are crucial in preemptive defense.

The WannaCry attack illustrated that many organizations were unprepared for ransomware attacks, emphasizing the importance of backup strategies and emergency response planning. The Sony Pictures hack brought to light the significance of guarding against insider threats and the need for robust data encryption and access controls. The SolarWinds breach demonstrated the complexities of securing a vast and interconnected digital ecosystem, highlighting the need for enhanced vigilance in third-party relationships and supply chain management.

These case studies collectively underline a key principle in modern cybersecurity: the need for a multi-layered defense strategy. This includes not only technological solutions but also organizational policies and a culture of security awareness. The evolving nature of cyber threats demands continuous adaptation and learning from past incidents to bolster defenses against future attacks.

The case studies highlight the critical need for a comprehensive cybersecurity approach, combining technology, policy, and awareness, with emphasis on incident response, international cooperation, and continuous learning to combat evolving cyber threats effectively.

Furthermore, these incidents emphasize the importance of incident response planning. Quick and coordinated responses can significantly mitigate the damage caused by cyber attacks. This includes having a clear communication plan, roles and responsibilities defined, and regular drills to ensure preparedness.

Lastly, the role of international cooperation and legal frameworks in combating cybercrime becomes evident. As cyber threats increasingly cross borders, collaboration among countries and the development of comprehensive legal measures are essential to deter and respond to these global threats effectively.

Analyzing these high-profile cyber attacks provides invaluable lessons for cybersecurity professionals. Understanding the methods employed in these attacks, the responses initiated, and their effectiveness is crucial for developing more robust cybersecurity strategies. This ongoing analysis not only prepares us for existing threats but also provides foresight into the nature of future cyber challenges.

2.1.4 Current Limitations in Cybersecurity

From the technical basics of protecting increasingly complex network infrastructures to the human factors that are often the weakest pillar of security chains, this section explores the obstacles that impede the progress of robust cyber security measures. Additionally, structural and legal frameworks that struggle to keep pace with rapid technological evolution highlight the necessity for a comprehensive and adaptive approach to cybersecurity. This introduction sets the stage for a detailed examination of these constraints and their implications for the future of digital security.

Technical Limitations

Cybersecurity faces technical hurdles like complex network infrastructures, quantum computing threats, and IoT device vulnerabilities. Human factors, including social engineering and human error, exacerbate security challenges. Additionally, the cybersecurity skills gap, outdated legal frameworks, and insufficient international cooperation hinder the development of effective cybersecurity measures, underscoring the need for a comprehensive and adaptive approach to digital security.

The field of cybersecurity is constantly evolving, but it faces significant technical challenges. One primary concern is the complexity of modern network infrastructures. With the increasing adoption of cloud services and distributed architectures, securing these vast and diverse networks becomes a formidable task. The interconnectivity of systems, while beneficial for efficiency and functionality, also increases the potential attack surface.

Another growing concern is the rise of quantum computing. Traditional cryptographic methods, which form the backbone of current digital security, may become vulnerable to quantum attacks[161]. Quantum computers, with their ability to solve complex mathematical problems much faster

than classical computers, could potentially break encryption algorithms that are currently considered secure.

The Internet of Things (IoT) presents another significant challenge. With billions of devices connected to the internet, ranging from home appliances to industrial equipment, ensuring the security of these devices is a daunting task[167]. Many IoT devices have inadequate security features, making them easy targets for cybercriminals. The diversity of IoT device manufacturers and the lack of standardized security protocols further complicate this issue.

Human Factors

Human factors play a critical role in cybersecurity. Social engineering attacks, such as phishing and pretexting, exploit human psychology rather than technical vulnerabilities. These types of attacks are becoming more sophisticated and are often the initial step in major data breaches.

Another human-related issue is human error. Mistakes made by employees, such as misconfiguring servers or falling for phishing scams, are a significant source of security breaches[207]. Even with the best technical safeguards in place, human error can provide an easy entry point for attackers.

The cybersecurity skills gap is also a pressing concern. The rapidly evolving nature of cyber threats requires a workforce with up-to-date skills and knowledge[290]. However, there is a global shortage of skilled cybersecurity professionals, which leaves organizations vulnerable to cyber attacks.

Structural Challenges

Organizational structures and legal frameworks play a crucial role in cybersecurity. Many organizations lack a clear cybersecurity strategy, and responsibilities are often not clearly defined. This can lead to gaps in security coverage and a lack of accountability when breaches occur[185].

The legal frameworks governing cybersecurity are often outdated and struggle to keep pace with the fast-evolving nature of technology and cyber threats. International cooperation is also a challenge, as cybercrime often crosses

national borders[114]. Without cohesive international laws and cooperation, tracking and prosecuting cybercriminals become increasingly difficult.

Future Outlook

Looking to the future, the cybersecurity landscape is expected to encounter new challenges. The increasing use of artificial intelligence (AI) and machine learning in cyber attacks could lead to more sophisticated and automated threats. The integration of AI in cybersecurity solutions is a potential countermeasure, but it also raises concerns about the AI arms race in cyber warfare.

Emerging technologies such as 5G, blockchain, and augmented reality will also bring new security challenges. Each of these technologies will require novel security approaches to address their unique vulnerabilities.

The future cybersecurity landscape will fight with challenges from AI-driven cyber threats and the integration of emerging technologies like 5G and blockchain, necessitating advanced, multifaceted defense strategies that encompass technological innovations, skilled workforce development, and robust organizational and legal frameworks to ensure comprehensive security and resilience against evolving cyber risks.

The current limitations in cybersecurity are multifaceted, encompassing technical, human, and structural aspects. Addressing these limitations requires a holistic approach, combining advanced technology, skilled professionals, robust organizational practices, and comprehensive legal frameworks.

Technical solutions need to evolve to address the complexity of modern networks, the potential threat from quantum computing, and the security challenges posed by IoT devices. This involves not only developing new security tools and protocols but also ensuring that existing systems are continuously updated and patched.

On the human front, increasing awareness and training can mitigate the risks of social engineering attacks and human errors. Organizations must invest in regular employee training and create a culture of security mindfulness. Additionally, addressing the cybersecurity skills gap is crucial. This could involve educational initiatives, professional training programs, and incentives to attract more talent into the cybersecurity field.

Structurally, organizations need to develop clear cybersecurity strategies with well-defined roles and responsibilities. This includes creating incident response plans and ensuring regular security audits and assessments. On a larger scale,

updating legal frameworks and enhancing international cooperation are essential for an effective global response to cyber threats.

Future research and development in cybersecurity should focus on anticipating and mitigating the risks associated with emerging technologies. This includes understanding the implications of AI, 5G, blockchain, and other advancing technologies on the cybersecurity landscape. Collaboration between industry, academia, and government will be key in driving innovation and preparing for future challenges in cybersecurity.

2.2 Artificial Intelligence: Evolution, Current State, and Challenges

Artificial Intelligence (AI) is a multidisciplinary field of computer science and engineering that focuses on creating intelligent systems capable of mimicking human-like cognitive functions, such as problem-solving, learning, reasoning, perception, and decision-making. AI aims to develop computer systems that can perform tasks that typically require human intelligence, ranging from simple tasks like language translation to complex activities like autonomous decision-making in self-driving cars[252].

AI is a multidisciplinary field creating intelligent systems mirroring human cognition. It analyzes data, makes quick decisions, and has diverse applications. The ultimate goal is achieving Artificial General Intelligence (AGI), akin to human intelligence, but it remains challenging.

The concept of AI is rooted in the aspiration to create machines that can simulate human intelligence and adapt to changing circumstances. AI systems are designed to analyze vast amounts of data[137], derive insights, and make informed decisions, often faster and more accurately than humans. This capability has led to the proliferation of AI across various domains, including healthcare, finance, transportation, and entertainment.

The ultimate goal of AI is to develop systems that can exhibit general intelligence, akin to human intelligence, by understanding context, adapting to new situations, and learning from diverse experiences. Achieving this level of AI, often referred to as Artificial General Intelligence (AGI) or Strong AI[97], remains a challenging and ongoing pursuit in the field.

2.2.1 History of Artificial Intelligence

The history of Artificial Intelligence (AI) begins with the visionary ideas of pioneers like Alan Turing and John McCarthy[36]. Turing, often considered the father of theoretical computer science and AI, proposed the concept of a universal machine capable of performing any conceivable mathematical computation. This idea laid the groundwork for modern computing and, by extension, AI. In 1950, Turing introduced the Turing Test, a method for determining whether a machine can exhibit intelligent behavior indistinguishable from a human.

Around the same time, John McCarthy, an American computer scientist, coined the term "Artificial Intelligence" in 1956[176], which he defined as the science and engineering of making intelligent machines. McCarthy's contribution to AI includes his work on Lisp, a programming language that became crucial for AI development due to its ability to handle symbolic information effectively.

The history of AI traces back to visionaries like Alan Turing, who proposed the universal machine concept, and John McCarthy, who coined "Artificial Intelligence" in 1956. Early AI efforts in the 1950s and 60s laid foundational work, but the field encountered setbacks, known as "AI winters," due to overhyped expectations and technical limitations. A resurgence in the late 1990s, driven by increased computational power and data availability, led to significant advancements, particularly in machine learning and deep learning, solidifying AI's impact across various industries.

The 1950s and 60s saw several key developments in AI. Early AI programs were relatively simple by today's standards. They included efforts like the Logic Theorist and General Problem Solver, which were designed to mimic human problem-solving and reasoning abilities. Though limited in their capabilities, these programs laid the foundation for future AI research.

In the 1970s and 80s, AI research began to focus on expert systems, which were programs designed to mimic the decision-making abilities of human experts. These systems were among the first successful commercial applications of AI, used in fields like medicine for diagnostic systems and in finance for fraud detection[133].

Despite early enthusiasm, AI faced significant challenges, leading to periods known as "AI winters"[187]. These were times when funding and interest in AI research waned, primarily due to inflated expectations and the subsequent disillusionment, along with technological limitations. The first AI winter occurred in the 1970s, triggered by the realization that the then-current AI technology was not capable of meeting the ambitious goals set by the field's pioneers.

A second AI winter occurred in the late 1980s and early 90s, partly due to the limitations of expert systems, which were

brittle and expensive to maintain. They were unable to adapt to changing environments or handle problems outside their narrow area of expertise.

The late 1990s and early 2000s witnessed a resurgence in AI, fueled by several key factors[65]. Increased computational power, made possible by advances in computer technology, allowed researchers to run more complex models. The advent of the internet and the digital age led to an explosion in the availability of data, often referred to as "big data," which provided the raw material needed to train increasingly sophisticated AI algorithms.

This period saw the development and refinement of machine learning algorithms, particularly deep learning, which leveraged large neural networks for tasks such as image and speech recognition with unprecedented accuracy. This era also marked the solidification of AI as a staple in various industries, setting the stage for the transformative impact it has today.

The history of AI is a tale of visionary ideas, technological challenges, periods of skepticism, and remarkable advancements[190]. It reflects a journey of understanding, developing, and harnessing one of the most revolutionary technologies in human history.

2.2.2 Recent Advances and Current Situation

In recent years, Artificial Intelligence (AI) has witnessed several groundbreaking developments that have propelled it to the forefront of technology and innovation. A key breakthrough in this domain has been the advent of deep learning. Utilizing large neural networks, deep learning has enabled significant advances in complex tasks like image and speech recognition. For instance, Google's DeepMind developed AlphaGo[48], an AI program that defeated a world champion in the ancient board game Go[16][103], a feat previously thought to be decades away.

Another major advancement is in Natural Language Processing (NLP). AI systems can now understand, interpret, and respond to human language with a remarkable level of sophistication. This progress is evident in technologies like OpenAI's GPT-3[17], which can generate coherent and

16: Go game is a strategic board game for two players, originating in China over 4,000 years ago, focusing on territorial control.

17: OpenAI's GPT-3 is an advanced AI language model capable of generating human-like text, answering questions, and performing diverse linguistic tasks.

contextually relevant text based on prompts, showcasing an impressive understanding of language nuances[168].

Reinforcement learning has also made strides, particularly in environments where AI must make a series of decisions leading to a specific goal. This approach, which involves learning optimal actions through trial and error, has been crucial in developing systems that excel in dynamic and complex environments, such as robotic navigation and automated trading systems[143].

AI Today

Today, AI is integrated across various sectors, revolutionizing how industries operate. In healthcare, AI algorithms assist in diagnosing diseases more accurately and quickly, such as analyzing X-rays and MRI images for signs of cancer[30]. AI-driven predictive analytics are used in personalized medicine to tailor treatment to individual patients.

In the financial sector, AI is employed in fraud detection, risk management, and algorithmic trading, where it analyzes large volumes of data to make informed decisions[98]. Banks and financial institutions use chatbots powered by AI to enhance customer service, providing 24/7 assistance and handling routine inquiries.

The automotive industry has been transformed by the advent of self-driving cars[252]. Companies like Tesla and Waymo are leading the charge, using AI to process data from various sensors and cameras to navigate safely[18].

18: Tesla and Waymo are companies specializing in autonomous driving technologies, with Tesla focusing also on electric vehicles and Waymo on self-driving systems mainly.

AI's role in customer service has expanded through the use of chatbots and virtual assistants. These AI-powered tools can handle a wide range of customer interactions, from answering frequently asked questions to providing personalized recommendations, improving efficiency and customer satisfaction[11].

In the realm of big data analytics, AI algorithms excel at extracting insights from vast amounts of data, facilitating decision-making in business strategies, marketing, and more[137]. AI is also pivotal in personalized content recommendations, as seen in platforms like Netflix and Spotify[106], where it analyzes user preferences to suggest relevant content.

Notable Outcomes

AI's impact is also significant in scientific research and environmental conservation. In drug discovery, AI algorithms accelerate the process of identifying potential drug candidates, reducing the time and cost involved in bringing new drugs to market. This was particularly evident in the rapid development of COVID-19 vaccines[134], where AI played a crucial role in analyzing viral protein structures.

AI significantly impacts various fields, from speeding up drug discovery and enhancing manufacturing efficiency to addressing climate change and aiding wildlife conservation. It accelerates vaccine development, improves precision in robotics, aids in climate modeling, optimizes energy use, and supports sustainable agriculture, showcasing its transformative potential across industries.

In manufacturing, AI-driven automation and predictive maintenance have enhanced efficiency and productivity. Robots equipped with AI perform complex tasks with precision, while AI systems predict equipment failures before they occur, reducing downtime and maintenance costs[183].

One of the most crucial contributions of AI is in addressing climate change. AI algorithms are used in climate modeling, helping scientists better understand climate patterns and predict future changes[62]. They are also integral in optimizing energy consumption in various industries and in the development of smart grids, which more efficiently manage and distribute renewable energy resources.

Moreover, AI has been instrumental in wildlife conservation efforts[101]. Through the analysis of data from satellite images and sensors, AI helps track animal populations and detect illegal activities, like poaching or deforestation, in protected areas.

In agriculture, AI assists in optimizing crop yields and reducing waste. By analyzing data from various sources, such as satellite imagery and soil sensors, AI provides farmers with insights into optimal planting times, crop rotation strategies, and pest control, leading to more efficient and sustainable farming practices[139].

The recent advances in AI have not only led to technological breakthroughs but have also had a transformative impact across various sectors. The integration of AI into everyday life and its contribution to solving some of the world's most pressing problems highlight its importance and potential. As AI continues to evolve, its role in shaping the future of industries and addressing global challenges becomes increasingly significant.

2.2.3 Machine Learning and Deep Learning

In this section, we will explore various aspects of machine learning, deep learning, and their applications, providing a solid foundation for understanding the topics discussed within this section.

Supervised, Unsupervised, and Reinforcement Learning

Machine learning paradigms offer diverse approaches to understanding and making predictions from data. Supervised learning leverages labeled data to train models on known input-output pairs. Unsupervised learning explores data to find inherent structures without pre-defined labels. Reinforcement learning interacts with environments to learn strategies based on rewards and penalties. Each paradigm has its distinct algorithms, models, training techniques, and data requirements, contributing uniquely to the advancement of artificial intelligence.

In this chapter, we look at the fundamental paradigms of machine learning: Supervised Learning, Unsupervised Learning and Reinforcement Learning. Each paradigm is distinguished by its unique approach to learning patterns from data, driven by the nature of the available information and the specific objectives of the learning process.

Supervised Learning stands as the most prevalent paradigm within the machine learning domain. It operates on the principle of learning a function that maps an input to an output based on example input-output pairs. This paradigm is characterized by its reliance on labeled datasets, where each example is paired with the correct output[73]. Some of the most commonly used and common Algorithms and Models[41] are as follows:

- Linear Regression: This algorithm models the relationship between a scalar dependent variable and one or more independent variables using a linear approach.
- Logistic Regression: Despite its name, logistic regression is used for binary classification problems, predicting the probability that an input belongs to a default class.
- Decision Trees: These models use a tree-like graph of decisions and their possible consequences, including chance event outcomes, resource costs, and utility.
- Support Vector Machines (SVM): SVMs are capable of performing linear and non-linear classification, regression, and outlier detection, particularly useful for complex intermediate-sized datasets.
- Neural Networks: These are networks of neurons either in software or hardware. Neural networks have been pivotal in handling vast arrays of structured and unstructured data.

To effectively develop and refine machine learning models, it's crucial to understand the various training techniques[294] and data requirements involved:

- Gradient Descent: This iterative optimization algorithm is used to minimize the cost function, a common technique for training a wide range of models, especially in deep learning.
- Backpropagation: In neural networks, backpropagation is a method used to calculate the gradient of the loss function with respect to each weight by the chain rule, updating weights in the opposite direction of the gradient.
- Cross-Validation: This technique assesses how the results of a statistical analysis will generalize to an independent dataset, particularly useful in scenarios where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice.
- Data Requirements: Supervised learning requires a substantial amount of labeled data. The quality and quantity of this data directly impact the model's performance, necessitating careful data collection, cleaning, and preprocessing.

Unsupervised Learning is the training of models to find patterns in data without explicit instructions on what to predict[151]. Models are exposed to vast amounts of data and tasked with identifying any patterns or structures within. The most widely used models and algorithms[206] are the following:

- K-Means Clustering: This algorithm partitions n observations into k clusters in which each observation belongs to the cluster with the nearest mean.
- Hierarchical Clustering: This method builds a hierarchy of clusters strategies include divisive (top-down) and agglomerative (bottom-up) approaches.
- Principal Component Analysis (PCA): PCA is a technique used to emphasize variation and bring out strong patterns in a dataset, reducing the dimensionality while retaining most of the variance.
- Autoencoders: These neural networks are designed to copy their input to their output, internally compressing the input into a lower-dimensional code and then reconstructing the output from this representation.

Exploring Unsupervised Learning, we examine various training techniques and the data requirements associated with each of them:

- Density Estimation: This involves constructing an estimate of the distribution that generated the dataset. It is a key step in the process of model selection[31].
- Anomaly Detection: Unsupervised learning is often used to detect unusual patterns that do not conform to expected behavior, known as outliers[38].
- Data Requirements: Unlike supervised learning, unsupervised learning does not require labeled data, making it more applicable to scenarios where obtaining labels is difficult or expensive.

Reinforcement Learning is a type of dynamic programming that trains algorithms using a system of rewards and penalties[274, 26]. Learning is achieved by trial and error, automatically determining the ideal behavior within a specific context, to maximize its performance. Here's a list of some notable reinforcement learning algorithms and methods:

- Q-Learning: This is a model-free reinforcement learning algorithm to learn the value of an action in a particular state[57].
- Deep Q-Network (DQN): Combining Q-Learning with deep neural networks, DQNs can approximate the Q-function, enabling the solution of complex reinforcement learning problems[83].
- Policy Gradient Methods: These methods optimize the policy directly by adjusting the parameters of the policy in the direction that increases the expected rewards[245].

To achieve optimal results in your training, it's essential to understand the various techniques and their corresponding data requirements, as outlined below:

- Exploration vs. Exploitation: Reinforcement learning models must balance the exploration of uncharted territory with the exploitation of current knowledge[59].
- Reward Function Design: The design of the reward function is critical in reinforcement learning, as it guides the learning algorithm towards the desired behavior.

- Data Requirements: Reinforcement learning does not require a dataset in the traditional sense but rather learns from interactions with an environment, which can be real or simulated.

Deep Learning Architectures

Deep Learning, a subset of machine learning, leverages neural networks with many layers (hence "deep") to model complex patterns in data. This chapter focuses on two pivotal deep learning architectures: Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), including Long Short-Term Memory (LSTM) networks, which have been instrumental in advancing fields such as image and speech recognition, natural language processing, and more.

The exploration of deep learning architectures, specifically CNNs and RNNs (including LSTMs), reveals the depth and breadth of their applicability across various domains. CNNs' ability to hierarchically process spatial data makes them indispensable in image-related tasks, while RNNs and LSTMs' proficiency in handling sequential data underpins significant advancements in NLP and speech recognition. These architectures continue to be at the forefront of deep learning research, driving ongoing improvements in both performance and efficiency across a multitude of applications.

Convolutional Neural Networks (CNNs) are specialized deep learning architectures for processing data that has a grid-like topology, such as images[155]. CNNs are inspired by the biological visual cortex where individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The architectural advantages of CNNs lie in their ability to automatically and adaptively learn spatial hierarchies of features from input images[212]. Some of the key Components are the following:

- Convolutional Layers: These layers perform a convolution operation that filters the input data, creating feature maps that summarize the presence of detected features in the input.
- Pooling Layers: Also known as subsampling or downsampling, these layers reduce the dimensionality of each feature map but retain the most important information.
- Activation Functions: Activation functions in CNNs, like ReLU or Sigmoid[19], introduce non-linear properties, enabling the network to learn complex patterns and decisions from data. They play a crucial role in deep learning models by allowing layers to capture intricate data relationships.

19: ReLU (Rectified Linear Unit) introduces non-linearity with a threshold at zero, while Sigmoid maps input values to a (0,1) range, facilitating binary classification.

Some of the architectural advancements that have raised these models to better achievements in image and grid-like data analysis are the following:

- Deep Architectures: Modern CNNs, such as AlexNet, VGG, GoogLeNet, and ResNet, have pushed the boundaries of depth, with architectures going much deeper than ever before, significantly improving image classification and object detection performances[13].
- Dropout and Batch Normalization: Techniques like dropout and batch normalization[91] have been pivotal in preventing overfitting[20] and speeding up convergence, respectively, in deep networks.

20: Overfitting occurs when a model learns the training data too well, capturing noise and outliers, which reduces its generalization to new data.

Now, let's explore the practical applications and use cases where CNNs have proven their effectiveness in various fields and tasks:

- Image and Video Recognition: CNNs have been the cornerstone of advancements in computer vision[10], enabling high-accuracy image and video classification, object detection, and image segmentation tasks.
- Medical Image Analysis: In healthcare, CNNs are used for diagnosing diseases from medical imaging data, such as X-rays, MRIs, and CT scans, by identifying patterns not discernible by the human eye[15].

Recurrent Neural Networks (RNNs) are a class of neural networks designed to handle sequential data. Unlike traditional neural networks, which assume that inputs and outputs are independent of each other, RNNs possess a memory that captures information about what has been calculated so far, making them ideal for tasks where context and temporal dependencies are crucial[221]. To Go deeper into the world of Recurrent Neural Networks (RNNs) and their specialized variant, Long Short-Term Memory (LSTM) networks, let's now explore the challenges associated with these powerful models[202].

- Vanishing and Exploding Gradients: Traditional RNNs are plagued by these issues, making training deep RNNs challenging. Long Short-Term Memory (LSTM) networks, a special kind of RNN, are designed to avoid these problems and are capable of learning long-term dependencies.
- LSTM Architecture: LSTMs introduce the concept of gates - input, output, and forget gates, along with a cell state, allowing the network to regulate the flow of information.

When defining the advanced variants of RNNs and LSTM we need to cite the following, which offer innovative solutions to address these challenges and push the boundaries data processing.

- Gated Recurrent Unit (GRU): GRUs are a simplified version of LSTMs with fewer parameters, combining the forget and input gates into a single "update gate," making them faster to train without significantly compromising the performance[67].
- Bidirectional RNNs and LSTMs: These networks consist of two RNNs/LSTMs that are trained on the input sequence in forward and reverse order. This structure allows the networks to have both backward and forward information about the sequence at every point.

And lastly, we will mention practical applications and use cases of recurrent neural networks (RNNs) and LSTMs:

- Natural Language Processing (NLP): RNNs and LSTMs are foundational in NLP for tasks like language modeling, text generation, and machine translation, where understanding the sequence and context of words is critical[52].
- Speech Recognition: In speech recognition, these networks excel by capturing the temporal dependencies of spoken language, translating audio signals into text[179].

Advanced Machine Learning Techniques

In this section, we will analyze some advanced machine learning techniques that have been revolutionizing the field in recent years. We will explore Transfer Learning, Generative Adversarial Networks (GANs), and AutoML, providing technical insights and practical applications for each.

Transfer Learning is a machine learning paradigm that leverages knowledge gained from one task to improve the performance of a related but different task. Instead of training a model from scratch for each new problem, transfer learning allows us to take advantage of pre-trained models and adapt them to our specific needs. This not only saves computational resources but also helps in situations with limited labeled data[237].

One common approach in transfer learning is to use the pre-trained model as a feature extractor. For instance, in computer vision, you can take a Convolutional Neural Network (CNN) trained on a large image dataset and use the activations of its layers as features for a new task[174]. By doing this, you can benefit from the CNN's ability to capture hierarchical and abstract features without training a new model from scratch.

Another technique is fine-tuning, where you start with a pre-trained model and train only a few of its top layers on your specific task[281]. This approach is particularly useful when you have a relatively large dataset for your task, allowing you to adapt the model to the unique characteristics of your data while preserving the knowledge learned from the pre-training.

Transfer Learning optimizes pre-trained models for new tasks, saving resources and aiding in data-scarce situations. GANs, through adversarial training, excel in generating realistic synthetic data for diverse uses. AutoML democratizes machine learning, automating the entire process to make sophisticated modeling accessible to a broader audience, thereby revolutionizing fields like healthcare, finance, and business analytics.

Transfer learning has found applications in various domains:

- Natural Language Processing (NLP): Transfer learning has improved the performance of sentiment analysis, named entity recognition, and text classification tasks by using pre-trained language models like BERT, GPT, and RoBERTa.
- Computer Vision: Image classification, object detection, and facial recognition tasks have benefited from transfer learning using architectures like VGG, ResNet, and MobileNet.
- Healthcare: Transfer learning has been used for medical image analysis, where pre-trained models are fine-tuned to detect diseases like cancer and diabetic retinopathy.
- Recommendation Systems: In e-commerce and streaming platforms, transfer learning helps improve user recommendations by leveraging data from similar domains.

Generative Adversarial Networks (GANs) were introduced in 2014 by Ian Goodfellow and colleagues[102]. GANs consist of two neural networks: a generator and a discriminator (Figure 2.5). The generator aims to produce data indistinguishable from real data, while the discriminator distinguishes between real and generated data. This adversarial training process results in the generation of high-quality synthetic data.

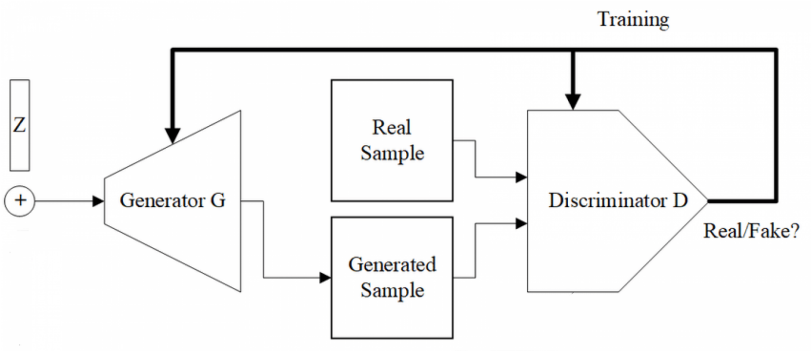


Figure 2.5: GANs train a "generator" to create new images from the latent representation of the source image, and a "discriminator" to evaluate the realism of the generated materials[79].

Several GAN architectures have emerged with unique strengths. The original GAN framework, known as Vanilla GAN, laid the foundation[21]. Deep Convolutional GANs (DCGANs) use convolutional layers for image generation[236]. Conditional GANs (cGANs) generate data based on specific inputs, enabling control over the generated output[47]. CycleGAN is designed for image-to-image translation tasks, such as turning photos into paintings or changing seasons in images[78]. StyleGAN focuses on generating highly realistic images by controlling both style and structure[132].

21: Vanilla GAN is the original architecture of the Adversarial Generative Networks, based on this architecture, different architectures such as DCGANs, cGANs, CycleGAN and StyleGAN have been developed.

GANs find diverse applications, including image generation for creative design and art[76], data augmentation for tasks with limited training data[248] (e.g., medical imaging and rare event prediction), enhancing image quality and resolution (e.g., satellite imaging and medical diagnostics)[266], and anomaly detection to identify abnormalities in data for cybersecurity and fraud detection[75].

AutoML, or Automated Machine Learning, automates the entire machine learning process, including data preprocessing, feature engineering, model selection, and hyperparameter tuning[112]. The goal is to democratize machine learning by making it accessible to individuals and organizations with limited machine learning expertise.

AutoML platforms employ techniques like Bayesian optimization, random search, and genetic algorithms to optimize hyperparameters for machine learning models, significantly improving their performance without manual intervention. These platforms also provide a selection of machine learning algorithms and architectures optimized for various tasks, automatically choosing the best model for a given dataset and problem[130]. This automation saves users the effort of experimenting with multiple models.

Furthermore, AutoML solutions simplify the critical aspect of feature engineering by automating feature extraction, transformation, and selection. This reduces the need for manual feature engineering expertise, making machine learning more accessible to a wider audience.

AutoML finds applications in various domains, but its efficiency in the field of IOT and Edge Computing is remarkable. In IoT and edge computing, AutoML facilitates deploying efficient machine learning models on devices with limited resources, enhancing real-time data analytics and decision-making[291]. By automating model optimization, AutoML ensures these models are compact yet powerful, suitable for the constrained computational capacities of IoT sensors and edge devices, thereby broadening the scope for immediate, on-device intelligence and responsive actions in various applications.

2.2.4 Challenges and Limitations

As AI continues to advance and integrate into various aspects of society, it raises several ethical concerns:

AI advancements raise ethical concerns like algorithmic bias and privacy issues, alongside technical challenges such as the opaque nature of deep learning and high computational costs. Practical issues include integration difficulties, a skills gap, and potential job displacement, underscoring the need for a collaborative approach to navigate AI's societal impact.

- Bias in AI Algorithms: One of the most significant ethical issues is the potential for bias in AI systems. This occurs when the data used to train AI algorithms contain inherent biases, leading to discriminatory outcomes[295]. For example, facial recognition technology has been criticized for lower accuracy rates in identifying individuals of certain ethnicities, raising concerns about fairness and equality.
- Privacy Issues with AI Data Handling: AI systems often require access to vast amounts of personal data to function effectively. This raises concerns about privacy and data protection. The handling and storage of sensitive information by AI systems must be governed by robust privacy laws and ethical guidelines to prevent misuse.
- Ethical Implications of Autonomous Systems: The deployment of autonomous systems, such as in warfare and self-driving cars, poses significant ethical dilemmas. In warfare, the use of AI-driven drones raises questions about the moral implications of machines making life-or-death decisions. In the case of self-driving cars,

the decision-making process in critical situations, often referred to as the "trolley problem," presents a complex ethical challenge[194].

Technical Challenges

The development and deployment of AI also face several technical challenges:

- Black-Box Nature of Deep Learning: Deep learning models, particularly neural networks, are often seen as "black boxes" because it can be challenging to interpret how they make decisions or predictions[43]. This lack of transparency can be problematic, especially in critical applications like healthcare or criminal justice, where understanding the reasoning behind a decision is crucial.
- Need for Large Amounts of Data: Many AI models, especially those based on machine learning, require large datasets for training. Obtaining such datasets can be difficult and expensive. Additionally, the quality and diversity of the data are critical in building effective and unbiased AI models.
- Computational Costs: Training complex AI models, particularly deep learning models, requires significant computational power. This can lead to high costs and energy consumption[22], making it challenging for smaller organizations to leverage advanced AI technologies.

22: OpenAI has projected that the cost of training large AI models will increase from $100 million to $500 million by 2030, with the cost of training a single model ranging from $3 million to $12 million.

Practical Issues

The practical implementation of AI in business and industry brings its own set of challenges:

- Integration with Existing Systems: Integrating AI into existing business systems and processes can be challenging. It often requires significant changes in infrastructure and workflow, which can be costly and time-consuming.
- Skills Gap: There is a growing skills gap in the AI field. The demand for professionals skilled in AI and machine learning far exceeds the supply, posing a challenge for businesses looking to adopt AI technologies. This gap also raises concerns about the broader workforce, as AI

continues to automate tasks traditionally performed by humans.

- Impact on Employment: The automation of jobs by AI systems is a significant concern. While AI can increase efficiency and reduce costs, it also has the potential to displace workers, particularly in sectors like manufacturing and customer service. This raises questions about the future of work and the need for policies to manage the transition and retrain workers.

While AI offers remarkable capabilities and potential benefits, it is also accompanied by a host of ethical, technical, and practical challenges. Addressing these challenges requires a collaborative effort among technologists, ethicists, policymakers, and businesses to ensure that AI develops in a way that is beneficial, fair, and sustainable for society.

Implementing AI in business faces challenges like integration complexities, a growing skills gap, and potential job displacement, necessitating significant adaptations and policy considerations.

2.3 Application of Artificial Intelligence in Cybersecurity

The application of Artificial Intelligence (AI) in cybersecurity represents a significant leap forward in the digital era, blending advanced computational techniques with evolving security strategies. This section of the thesis offers a comprehensive literature review, examining key studies and articles that have explored AI's integration into cybersecurity.

2.3.1 Overview of Research and Articles

The concept of leveraging AI in cybersecurity traces back to foundational research which identified the potential of machine learning and AI-driven threat intelligence in enhancing digital security. Early indications of this potential are found in academic papers from the late 1990s and early 2000s, where researchers began exploring the use of machine learning algorithms for anomaly detection in network systems[87]. These studies laid the groundwork for the current landscape of AI in cybersecurity, focusing on the development of algorithms capable of identifying patterns and irregularities indicative of cyber threats.

One seminal paper in this area is T. Lane et al.'s 1997 study[146], "An Application of Machine Learning to Anomaly Detection", which provided early insights into how machine learning could be applied to identify unusual patterns in network data, signaling potential security breaches. This research marked a pivotal shift from traditional rule-based security systems to more dynamic, adaptive AI-driven models.

Key Studies and Findings

Since then, numerous studies have emerged that investigate the role of AI in cybersecurity. Some of the most important research includes:

Research in AI's role in cybersecurity spans malware detection, phishing prevention, and the utilization of various AI techniques, addressing challenges like unbalanced datasets, concept drift, and the rapid evolution of threats, advocating for enhanced automation and security measures.

Malware Detection: Gibert et al. review AI in malware analysis, categorizing techniques as static or dynamic, detailing feature extraction methods, AI models, and how malware evades AI, addressing challenges like unbalanced datasets and concept drift[95]. Shaukat et al. extensively review ML and DL applications in cybersecurity, covering spam, intrusion, and malware detection fields[232]. Aslan and Samet review malware classification, focusing on detection methods and challenges with advanced malware, detailing features and repositories used[22].

Phishing Attempts: Giovanni et al. analyzes phishing detection methods, highlighting their strengths and weaknesses, particularly regarding false positives and negatives, and discusses the limitations of current approaches[17]. Giovanni also discusses phishing prevention limitations in other research[18], focusing on detecting compromised webpages and alerting users to mitigate threats before they succumb to phishing attacks. Houssain et al. provides an overview of cybersecurity threats, giving weight to phishing as one of the most critical aspects. Emphasizing the rapid evolution of threats, the role of nation-state actors, and the lag in organizational defense capabilities, advocating for increased automation and security adoption[136].

Utilization of Various AI Techniques: The breadth of AI techniques applied in cybersecurity is vast, ranging from basic machine learning models to complex deep learning and neural network architectures. The study by Apruzzese et al. on the effectiveness of machine learning in intrusion

detection systems exemplifies the diverse application of these techniques in cybersecurity[16].

Many online services rely on the interaction between client and server systems. Attackers may interfere with these communications, blocking server access or hindering the server's ability to respond to client inquiries, as seen in DoS (Denial of Service) attacks. In the case of a botnet, attackers initially infect multiple hosts (via Trojans or other malware forms) to take command and direct them to perform specific actions. For example, during a DoS attack, these infected hosts might flood a server with excessive requests, depleting its capacity to service legitimate user requests.

Online services are vulnerable to Denial of Service (DoS) attacks, where attackers use botnets to overwhelm servers with excessive requests, hindering legitimate operations. Research has developed methods to detect DoS attacks, particularly from IoT devices, using network behavior features with high accuracy. Emerging technologies, like smart meters and Software-Defined Networking (SDN), face novel DoS threats. Studies utilizing AI, including deep learning, have shown effectiveness in detecting DoS attacks in SDNs with high accuracy. However, challenges remain in real-time detection and addressing application-layer attacks, underscoring the need for advanced solutions.

Denial of Service (DoS) attacks are escalating in severity as the botnets powering them become more complex and extend across various platforms, including PCs, smartphones, and IoT devices. A study successfully identified DoS attacks initiated by IoT devices through the utilization of distinct features that encapsulate IoT network behavior [72]. It was noted that IoT devices typically interact with a constrained set of endpoints during application operation. To capture this characteristic, the study introduced two key features: a) the count of unique destination IP addresses, and b) the count of unique IP addresses observed within a brief, 10-second interval. Additional attributes considered were the time intervals between packet arrivals, alongside the rate of change in these intervals over time, pinpointing an abrupt surge in traffic originating from an IoT device. The findings demonstrated that using decision trees for analysis yielded a detection accuracy of 99 percent. Given that IoT devices generally connect through a singular gateway, like a household router, the implementation of this detection technique at gateway level offers a promising approach to thwarting DoS attacks from IoT sources.

As emerging technologies introduce new services, they also become targets for novel DoS attack methodologies. A notable instance is the series of DoS attacks targeting smart meters, as documented in recent studies [283, 227]. These smart meters not only measure utility usage but also function as nodes within a mesh network of similar devices. In study [283], researchers discovered that the introduction of a malicious packet into a single meter could trigger an overwhelming flood of routing packets. This deluge forces other meters within the network to alter their routing data, ultimately

blocking the flow of legitimate data packets to their intended destinations. Consequently, the network's meters tirelessly reroute, striving to deliver the data packet, which leads to network paralysis. Meanwhile, study [227] highlighted the susceptibility of smart meters' wireless components to jamming attacks. The detection method proposed involved assessing the variance in the signal distance from what's calculated as the network's centroid. With the continual advent of new services and computational frameworks, the complexity and sophistication of DoS attack vectors are anticipated to evolve correspondingly.

Recent research [193, 148, 249] has concentrated on identifying DoS (Denial of Service) attacks within environments managed by Software-Defined Networking (SDN). SDN's approach to network management significantly diverges from the operational mechanisms of traditional routing protocols. Unlike conventional routers, which direct traffic based on static routing tables, SDN employs a dynamic strategy that involves the collection and programmatic analysis of network data prior to the forwarding of network traffic. This distinction introduces a unique set of challenges for detecting DoS attacks in SDN contexts [282]. The investigation detailed in [227] developed 68 distinctive features based on packet data managed by the SDN's data plane before being relayed to the control plane. These features, derived from the analysis of packet statistics such as ratio, entropy, count, size, flow, and flags within the Internet Protocol (IP), Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Internet Control Message Protocol (ICMP) packets, were instrumental in the process. Employing deep learning techniques, this study achieved a 95.65 percent success rate in pinpointing DoS attacks.

Deep Learning has been recognized as a highly effective method for identifying DoS attacks within Software-Defined Network (SDN) environments. The researchers in study [148] utilized a set of 20 distinct features, including protocol type, port numbers, and packet sizes. They demonstrated that a variant of Deep Learning known as Long Short-Term Memory (LSTM) achieved a remarkable 99.88 percent success rate in detecting DoS attacks. Meanwhile, the analysis presented in [249] leveraged various metrics such as the count of connections over a two-second interval, the length of these connections, and the volume of data exchanged in each di-

rection. This study illustrated that Deep Neural Networks (DNNs) surpassed traditional AI models like Support Vector Machines (SVMs), Naïve Bayes, and Decision Trees in accuracy. This superior performance of DNNs is attributed to their ability to generate hidden or latent variables, effectively acting as additional features, a capability not present in other machine learning approaches.

SDN benefits from AI methodologies to navigate the evolving computing landscape, drawing on historical network data to decipher new traffic patterns and forecast security tendencies. Nonetheless, the literature reveals two primary gaps in employing AI for cyberattack detection on SDNs. Firstly, the application of AI for real-time detection remains unexplored. The necessity for immediate analysis to distinguish between benign and malicious traffic underscores the challenge, as AI solutions typically undergo numerous computational iterations to yield a final verdict. Although research [148] explored the real-time efficacy of their system, this assessment occurred post the development of a classification model based on training data. To our knowledge, no existing research has introduced an AI-based approach tailored for SDN that facilitates instant DoS attack detection. Secondly, SDNs intrinsically do not cater to the identification of application-layer attacks. The scrutiny of DoS attacks targeting application-layer protocols mandates either an in-depth packet inspection or alternative decentralized methods.

Addressing the topic of application-layer security, where servers host critical organizational applications, targeting these servers presents a opportunity for attackers to compromise either the organizations or their users. Traditionally, attacks at the application layer targeted protocols such as HTTP, DNS, or SIP. With the introduction of HTTP/2, a new web communication protocol, innovative DoS attack strategies and detection methods were developed [2]. These strategies exploited HTTP/2's unique flow-control feature, a mechanism absent in the older HTTP/1.1, to overwhelm servers with a minimal number of connections, thereby eluding conventional intrusion detection systems that typically flag a high volume of connections as potential attacks [208]. The use of artificial intelligence techniques like Naïve Bayes, Decision Trees, and Rule Learning for detecting HTTP/2 flood attacks revealed a higher incidence of false positives compared to their application in identifying HTTP/1.1 DDoS

assaults, indicating their capability to circumvent established intrusion detection frameworks. SVMs, in particular, demonstrated remarkable efficiency in detecting such attacks without any false alarms, given a specific set of HTTP/2 detection features.

The evolution of application-layer attacks has extended from targeting server protocols like HTTP to altering information on social networks, notably affecting national security and public perception through misinformation. Advances in AI have shown promise in detecting these threats, with methods like SVMs effectively identifying novel attack strategies without false positives. However, the detection of misinformation, such as fake news, requires sophisticated AI techniques, achieving varied success across different platforms. Despite progress, the necessity for human intervention in refining detection methods remains, highlighting a blend of technological and human expertise in addressing cybersecurity challenges.

The landscape of application-layer attacks has evolved, shifting focus from hindering information flow to altering the meaning of information. This transformation is evident with the rise of online social networks and the emergence of cyberattacks aimed at spreading misinformation to manipulate public perception or decision-making processes [259]. A notable instance of this was the propagation of fake news during the 2016 US presidential election, which had significant implications for national security [230]. Misinformation poses a threat not only to national security but also to individual well-being, manifesting in various forms such as fake news, cyberbullying, and online grooming. The challenge of identifying and mitigating false information represents a contemporary issue in application-layer cybersecurity.

Artificial intelligence has emerged as a potent tool in combating misinformation, capable of analyzing vast datasets swiftly [96, 240, 66]. For instance, one study [96] scrutinized a dataset comprising 11,000 articles, identifying 29 percent as fake, and achieved a classification accuracy of 77.2 percent using Stochastic Gradient Descent. Another investigation [240] employed correlation-based classifiers to analyze over 150,000 tweets, demonstrating a significantly enhanced precision in message classification. Similarly, an analysis of 4.4 million Facebook messages utilizing algorithms like Naïve Bayes and RandomForest achieved an 86.9 percent accuracy in distinguishing between legitimate and fake messages [66].

Prompt detection of fake news is crucial, as demonstrated by a study [164] that introduced an early detection method using Artificial Neural Networks (ANNs), including Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), to analyze the timing and structure of news propagation. This approach achieved an accuracy of 85 percent on Twitter and 92 percent on Sina Weibo within minutes of the initial fake news post.

The fight against misinformation also leverages linguistic analysis to enhance text classification for cybersecurity [60].

Techniques have been developed to detect threats and authenticate user identities on platforms like Twitter by analyzing linguistic features such as grammar and word choice [96, 240, 217, 149, 77]. These methodologies have proven effective in improving public well-being and showcasing the potential of AI to incorporate novel features into cybersecurity measures.

Text classification has benefited from the tf-idf feature, which is augmented by other linguistic indicators for improved detection of false information [240, 217, 23]. SVMs have been used to identify satirical or potentially misleading news, while Naïve Bayes classifiers have proven effective in topic classification for spam and phishing detection on Twitter [80]. Deep Neural Networks (DNNs) have also shown their capability in identifying hate speech with high accuracy [23].

Despite these advancements, cyberattack detection at the semantic level remains a nascent field. Certain studies have underscored the necessity for human input in selecting relevant terms for threat detection [240, 217, 149, 77, 80]. Additionally, research exploring non-linguistic features for fake news detection on social media platforms highlights the importance of unique social media attributes, such as the presence of URLs in tweets, the ratio of followers to followees, and the timing of posts [28, 186]. This underscores the ongoing requirement for human expertise in cybersecurity efforts, alongside the integration of AI technologies.

In the realm of cybersecurity, the most significant vulnerability often lies with the individual users of the internet. These users typically concentrate on their immediate tasks rather than on the continuously evolving and expanding cyber threats. While technological solutions can be adapted to address known cyber risks, educating humans on cybersecurity requires ongoing and updated training. This necessity is a critical factor in the widespread dissemination of malware via contemporary phishing techniques [46].

Malware refers to software designed with harmful intentions, such as viruses, Trojans, or worms. Phishing is a tactic aimed at deceiving users into performing actions that benefit attackers, like clicking on malicious links or files, which may result in the distribution of malware or the disclosure of confidential information. Early phishing strategies exploited

vulnerabilities in human sensory perception with counterfeit emails or websites [68], making it difficult for individuals to differentiate them from genuine sources. Present-day phishing methods have evolved to exploit the limitations of human knowledge, requiring users to verify the authenticity of their targets, often by examining the underlying code of links [9], a task that may necessitate specific skills. Artificial Intelligence (AI) presents an opportunity to enhance human capabilities in this regard.

In cybersecurity, the primary vulnerability lies with users, often focused on tasks over evolving cyber threats. Technology can adapt to these risks, but user education on cybersecurity remains crucial, especially against malware and phishing, which exploit human vulnerabilities. AI enhances detection capabilities, like identifying phishing via predefined indicators with high accuracy. However, attackers now leverage JavaScript and social media, necessitating sophisticated knowledge for detection. Innovative solutions, like educational games, aim to improve awareness but raise privacy concerns.

AI can simplify the process of detecting phishing attempts by using predefined rules as indicators. One study [182] utilized Support Vector Machines (SVMs) to identify fraudulent banking website links. This method analyzed five indicators: the IP address, the presence of a Secure Sockets Layer (SSL) certificate, the number of dots in the URL, the length of the web address, and the presence of blacklist keywords. Authentic banking sites typically display a legitimate domain name rather than an IP address, have an SSL certificate, feature shorter URLs, and avoid using subdomains (which would increase the dot count). Additionally, this technique gathered terms frequently found on phishing sites. The findings indicated that this approach could accurately identify new phishing attempts with 98.86% precision, showcasing how AI training can mitigate the human vulnerability to cybersecurity threats.

The exploitation of human susceptibilities continues to be a common tactic among attackers, including those targeting contemporary websites and social media platforms. Modern sites enhance user experience through JavaScript, which improves interactivity and response times. However, attackers can use JavaScript to embed malware or execute phishing attacks. Identifying websites compromised by malicious JavaScript requires sophisticated coding knowledge, rendering detection virtually impossible for the average user. Moreover, new strategies involve spreading malware via social media by enticing users to click on malicious links, leading to unintended malware downloads (also known as drive-by downloads) [124]. AI has been applied to identify malicious JavaScript on websites [233, 284] and to detect drive-by downloads [6]. These AI techniques analyze various elements, such as the size of JavaScript words, character distribution in code, string bytecodes frequency, commenting patterns, and the use of sensitive functions, thereby surpass-

ing human limitations in recognizing and understanding such characteristics. Additionally, AI has been deployed to identify disguised harmful JavaScript [157] and implement safety measures to prevent malware proliferation following phishing incidents [265].

In the sphere of user-friendly security, the objective is to develop secure systems that are also accessible to the general public. One innovative approach to enhancing the average user's awareness of cybersecurity involves interactive games [19]. These games are designed to increase users' alertness to deceptive URLs that mimic legitimate ones, such as distinguishing between a counterfeit URL "`http:/www.amazonn.com`" and the genuine "`http:/www.amazon.com`". A review of 28 studies on cybersecurity training games revealed that although participants enjoyed the games, the research did not conclusively demonstrate their effectiveness [113]. The studies often involved small, selected groups of participants and did not adequately compare the cybersecurity awareness between players and a control group. Additionally, concerns have been raised regarding the privacy and trustworthiness of such educational games [33]. These games rely on algorithms that analyze users' confidence, attitudes towards software updates, creation of strong passwords, recognition of malicious links, and appropriate hardware usage (e.g., data backup). If adversaries were to access this information, it could be exploited to craft personalized phishing attacks. The risk intensifies if such data becomes publicly accessible or falls into unauthorized hands, raising significant privacy and trust issues [51].

With the widespread use of Internet-connected devices, which are becoming increasingly compact, their capability to gather data now significantly exceeds the human capacity to monitor such data collection processes. These devices harvest data like audio input, location information, environmental temperatures, and levels of ambient light to enhance the experience of the user. Nevertheless, research [140, 90, 54] indicates that the acquisition of such data could potentially be exploited for harmful purposes. Intelligent virtual assistants, including Amazon's Alexa, Apple's Siri, and Google Home, could be manipulated to unauthorizedly unlock a smart garage door or to covertly record confidential conversations [54]. One investigation [140] revealed that these devices could be employed for nefarious activities such as identifying

a covert location in an airport, engaging in cyberbullying, inciting fear, or redirecting a user's online navigation to display targeted advertisements. Furthermore, these devices might be used for illicit tagging of locations or individuals in connection with criminal activities [90].

The integration of Internet-connected devices into daily life enhances user experiences but raises significant privacy and security concerns. These devices, capable of collecting vast amounts of data, can potentially be exploited for malicious purposes. Traditional privacy safeguards, such as encryption, are becoming less effective with the advent of IoT. Innovative AI technologies and blockchain are now being leveraged to enhance privacy and security. AI is utilized in securing data pathways and recognizing abnormal device behavior, while blockchain offers a decentralized approach to data storage, enhancing privacy without central oversight. These advancements aim to address the challenges of ensuring data privacy in an increasingly connected world.

In the past, strategies to safeguard privacy focused on secure authentication methods, like encryption and the use of security certificates. However, with the advent of the IoT, where devices are often mobile and data is stored on cloud servers, these traditional mechanisms become less effective. AI technologies have been utilized to ensure the privacy of communications, especially when data pathways are subject to change and data is held by third-party servers. For instance, learning automata were leveraged to distribute security certificates among mobile vehicles [145], and artificial immune system algorithms were employed to autonomously secure Wireless Sensor Network (WSN) connections for mobile devices [220]. Given the dynamic nature of WSNs, where IoT devices frequently connect and disconnect from the network, conventional security protocols like port security become obsolete. Hence, in [220], the researchers introduced metrics such as packet reception rate, packet mismatch rate, and energy consumption per packet to monitor device behavior, using artificial immune systems to distinguish between normal and abnormal patterns. Detection of abnormal activity led to the dropping of unencrypted packets, highlighting the necessity for novel privacy techniques amidst the proliferation of Internet-connected devices. Additionally, the extensive storage of data on cloud platforms raises concerns about the potential for cloud operators to access sensitive information. To mitigate this risk, smart algorithms were devised to distribute data across multiple cloud servers [154], complicating unauthorized access by cloud personnel.

Biometrics and metrics analyzing human behavior have traditionally been incorporated into secure authentication systems. Yet, these systems often face difficulties in adapting to different operational conditions. AI methodologies, such as Genetic Algorithms, have been applied to enhance the reliability and accuracy of facial, fingerprint, and voice recognition systems across varied environments [188].

Blockchain technology presents a revolutionary approach to privacy enhancement, circumventing traditional legislative

constraints [189]. It enables a decentralized network of computers to encrypt and store data without the need for a central overseeing authority. AI has been integrated with blockchain technology [200, 173, 280] to enhance the security and efficiency of blockchain applications. For instance, in [200], AI was utilized to ensure secure communication between two IoT devices via blockchain, eliminating the reliance on centralized systems. This was achieved by employing Reinforcement Learning to evaluate if the data exchanged met the access control requirements of the devices, thereby facilitating autonomous resource sharing.

The study in [173] explored the use of blockchain in the healthcare industry to securely gather medical data for disease prediction and privacy preservation. The need for extensive data for classification and prediction algorithms poses a conflict with patient privacy interests. Blockchain technology allows for the secure recording of medical data, empowering patients with privacy assurances and control over their personal information, including access permissions. This not only fosters patient trust in digital data storage but also facilitates the use of personal health indicators for medical analysis. AI methods, such as Deep Neural Networks (DNNs), were applied to extract relevant features from medical imaging before blockchain recording, while Recurrent Neural Networks (RNNs) were used for chronic condition identification and disease forecasting.

In [280], AI strategies like similarity learning were applied in a smart contract-based data trading platform to address discrepancies between the data provided and the actual data received. This use of similarity learning to assess the congruence between the features of the data from both the purchaser and the provider underscores the evolving role of AI in addressing legal, regulatory, and ethical considerations in data privacy, highlighting its potential to positively impact human welfare through the ethical sharing of personal data.

Critical infrastructures are essential for the foundation of national security and societal well-being, encompassing sectors like energy (including oil, gas, electric, and nuclear), water supply, air traffic management, and telecommunication networks [5]. Protecting these critical infrastructures is crucial as the daily functioning and safety of the populace

hinge on their continuous operation and integrity. Discussions have evolved to reveal that the domain of cybersecurity extends beyond detecting intrusions in networks to potential enhancements in human welfare, propelled by various sectors including healthcare and education. Additionally, the sector focusing on crtical infrastructure has significantly contributed to the advancement of AI methodologies for bolstering cybersecurity measures.

Critical infrastructures, crucial for national security and societal well-being, include sectors like energy, water supply, air traffic, and telecommunications. Their protection, particularly through cybersecurity, is vital for uninterrupted operation and safety. The advancement of AI in cybersecurity offers significant improvements, such as predicting malfunctions and detecting anomalies, thereby enhancing resilience against cyber threats. AI technologies, including Artificial Neural Networks and Support Vector Machines, are employed for monitoring and controlling access to critical systems. Additionally, logical and mathematical models are developed for secure system access, autonomous restoration of communication channels, and fault rectification in infrastructures, showcasing the importance of AI and mathematical approaches in safeguarding critical infrastructures.

In the realm of critical infrastructures, cybersecurity is predominantly linked with the protection of SCADA systems, which serve as the primary control mechanisms of these infrastructures through a network of computing nodes [4]. These systems, located within the Operational Technology (OT) networks of organizations, face increasing risks of cyber threats as OT and Information Technology (IT) networks integrate more closely and connect to the internet.

Despite these challenges and vulnerabilities, it is imperative for critical infrastructures to withstand cyber threats to ensure uninterrupted business operations [5]. Enhancing the resilience of SCADA systems against disruptions can be achieved through the application of AI technologies. For instance, the prediction of malfunctions in wind turbine generators is made possible through the use of Artificial Neural Networks (ANNs), which monitor variables such as ambient temperature, generator velocity, and the pitch angle of generator power outputs [29]. In water management systems, AI methodologies like k-NN, Decision Trees, and SVMs are utilized for identifying various anomalies, including cyber assaults and equipment malfunctions [115]. AI technologies, specifically SVMs and ANNs, are also employed in regulating access to SCADA systems, taking into account dynamic user attributes like location, usage timing, and the user's shift schedule [293]. The continuous exploration of AI in enhancing the resilience of critical infrastructures is driven by their critical role in society.

Moreover, propositional logic-based AI approaches have been adopted for the security of critical infrastructures. A notable example is the development of a logic-based framework designed to implement security protocols for system access in SCADA environments, addressing the complex relationship between user privileges and system regulations [218]. This framework facilitates the distribution of rules across system nodes, enabling the derivation of permissible actions for a

user on each node. Upon a privileged user issuing a command to a node, the command along with the user's privilege data is sent to an authorization server. The server then processes this information, generates a token, and dispatches it along with the command to the targeted node, which makes the final decision on the command's execution based on its local authorization policy. This logic-based model ensures scalable security in SCADA systems by decentralizing authorization decisions.

Additionally, intelligent algorithms based on logic have been proposed to autonomously restore communication channels in SCADA systems [126]. These systems maintain secure communication through session keys. If a node fails, immediate restoration of its communication channel is crucial to prevent unauthorized control. The authors in [126] suggest distributing materials necessary for re-keying to remote nodes, enabling them to autonomously generate new session keys using a mathematical formula, thereby ensuring the system's self-reliance in maintaining secure communications.

Furthermore, mathematical models have been applied to automatically rectify faults in electrical distribution systems [44]. Following a fault, the self-healing mechanism identifies which network zone to isolate by evaluating a set of features like the cost of power losses and the voltage magnitude at each node. By employing set theory to group these features and then analyzing them with mathematical models representing the electrical distribution systems' steady state, these approaches underline the broad application of logic and mathematical techniques in fulfilling the cybersecurity needs of critical infrastructures.

Diverse Perspectives

The application of AI in cybersecurity has been explored from various perspectives:

- Academic Research: Academic studies have primarily focused on the theoretical and technical aspects of AI applications in cybersecurity, exploring novel algorithms and models.
- Industry White Papers and Reports: Industry contributions, through white papers and technical reports, provide a practical viewpoint, showcasing real-world

applications and performance metrics of AI in cybersecurity. For instance, reports from cybersecurity firms like Symantec and McAfee offer insights into the deployment of AI in combating real-time cyber threats.

- Case Studies from Cybersecurity Firms: Case studies from leading cybersecurity firms have been instrumental in illustrating the practical effectiveness and challenges of AI in this field. These studies often provide a detailed analysis of specific cyber-attack scenarios and how AI-driven tools were employed to mitigate these threats.

AI in cybersecurity is explored through academic research, industry reports, and case studies, highlighting theoretical, practical, and real-world applications and challenges.

The review of existing literature reveals a dynamic and rapidly evolving field where AI's capabilities are being harnessed to fortify digital security. From theoretical research to practical applications, the convergence of AI and cybersecurity is proving to be a pivotal development in the ongoing battle against cyber threats. This section sets the stage for a deeper exploration into the impact, efficacy, and future potential of AI in the realm of cybersecurity.

2.3.2 Impact and Efficacy of AI in Cybersecurity

This section analyses the impact and effectiveness of Artificial Intelligence (AI) in the field of cybersecurity. It aims to analyze how AI has reshaped cybersecurity practices and assess its efficacy through various statistics and data.

AI's integration into cybersecurity has revolutionized threat detection and response, introducing automated and predictive capabilities. Statistical evidence shows AI enhances detection accuracy and response times, reducing costs. Case studies, like Darktrace's ransomware prevention and JPMorgan Chase's network monitoring, exemplify AI's practical effectiveness in various sectors.

Impact of AI in Cybersecurity

The integration of AI into cybersecurity has brought about a paradigm shift in how cyber threats are detected, analyzed, and neutralized. AI's capability to process and analyze vast amounts of data at unprecedented speeds has significantly enhanced threat detection capabilities.

- Enhanced Threat Detection and Response: AI algorithms, particularly those based on machine learning, have enabled the development of advanced threat detection systems. These systems can identify subtle, anomalous patterns indicative of cyber threats, which might be missed by traditional security measures[23].

23: Traditional tools like IDS/IPS systems, implement AI-driven tools to enhance anomaly detection, automate threat responses, and improve accuracy, significantly reducing false positives and adapting to evolving cyber threats.

For instance, AI-driven tools are now capable of continuously monitoring network traffic and identifying deviations that could signal a breach[34].

- Automated Security Protocols: AI's role in automating responses to security incidents is another crucial area of impact[105]. Upon detection of a potential threat, AI systems can automatically initiate protocols to isolate affected networks, analyze the nature of the attack, and implement measures to prevent further damage.
- Predictive Capabilities: AI has introduced predictive capabilities in cybersecurity. Leveraging historical data, AI models can predict potential vulnerabilities and future attack vectors[35], allowing organizations to fortify their defenses proactively.

Statistical Analysis of AI's Efficacy

To understand AI's effectiveness in cybersecurity, it is essential to look at relevant statistics and data:

- Reduction in Detection and Response Times: According to a 2020 report by the Ponemon Institute, organizations using AI in cybersecurity reported a significant reduction in the time taken to detect and respond to threats[120]. The study noted that AI-enabled systems could reduce response times by up to 12% compared to traditional methods.
- Accuracy in Threat Detection: Research conducted by MIT's Computer Science and Artificial Intelligence Laboratory (CSAIL) demonstrated that AI systems could detect 85% of cyber attacks[191], a rate higher than traditional software-based approaches[24].
- Cost-Effectiveness: A study by Capgemini Research Institute revealed that 64% of organizations found that implementing AI in cybersecurity reduced the cost associated with detecting and responding to breaches[39].

24: The tool developed is called AI2 and predicts attacks by analyzing data with unsupervised machine learning to identify patterns and clustering suspicious activity, which is then verified by human analysts, refining the model's accuracy for future predictions.

Case Studies

To illustrate the impact of AI in real-life situations, there are numerous cases over the last years. To show a bunch of examples, I will discuss two relevant use cases, but there are a large number of use cases.

- AI in Combating Ransomware: AI-driven cybersecurity firm Darktrace reported successfully using machine learning algorithms to identify and stop a ransomware attack in its initial stages for a client[63], preventing substantial data loss and financial damage.
- AI in Financial Sector Cybersecurity: JPMorgan Chase's adoption of AI for cybersecurity purposes serves as another example[25]. The company employs AI to monitor network traffic and detect anomalous patterns, significantly reducing the incidence of false positives in threat detection.

25: JPMorgan annually invests $15bn in technology, employing 62,000 technologists mainly to fight cybercrime

The impact and efficacy of AI in cybersecurity are evident in its enhanced threat detection capabilities, automated response systems, and predictive analytics. The statistical data and case studies reinforce AI's role as a game-changer in the field, offering more efficient, accurate, and cost-effective solutions compared to traditional cybersecurity methods. This analysis not only underscores the current value of AI in combating cyber threats but also sets the stage for exploring its future potential and development trajectories in the cybersecurity domain.

2.3.3 Limitations and Future Lines of Research

In examining the application of Artificial Intelligence (AI) in cybersecurity, it becomes evident that while AI has brought significant advancements, it is accompanied by distinct limitations that necessitate further research. These challenges, inherent in the current state of AI in cybersecurity, present both obstacles and opportunities for future exploration and development.

Current Limitations of AI in Cybersecurity

AI's effectiveness against zero-day attacks, which are newly discovered vulnerabilities or threats not yet known in the cybersecurity world, stands as a notable limitation. These attacks are particularly challenging for AI systems, as they rely heavily on historical data and previous patterns to detect threats. The lack of prior knowledge or data on these novel threats limits AI's ability to predict and mitigate them effectively.

The performance of AI in cybersecurity is deeply tied to the quality and quantity of available data. AI models require extensive and diverse datasets to learn and make accurate predictions. However, in the realm of cybersecurity, gathering such data is fraught with challenges. Privacy concerns and the sensitive nature of cybersecurity data often restrict the availability of large-scale, diverse datasets, leading to potential biases or gaps in AI training.

The prospect of AI being used for malicious purposes also raises significant concerns. As AI technology becomes more accessible, there is an increasing risk of it being leveraged by cyber attackers. This could lead to the development of more sophisticated cyber threats, including malware that can adapt to counteract AI-driven security measures or the automation of large-scale cyber attacks. The duality of AI as both a tool for defense and a potential weapon for attackers necessitates a careful approach to its development and deployment in cybersecurity contexts.

AI's effectiveness in cybersecurity is limited by its reliance on historical data, making zero-day attacks challenging to counter. Data quality and privacy issues hinder AI training, while the potential for AI's malicious use and the high costs of implementation pose significant challenges, disproportionately affecting smaller organizations.

The complexity and cost of implementing AI-based solutions in cybersecurity present additional challenges. Establishing effective AI systems requires not only significant financial investment but also expertise in both AI and cybersecurity domains. For many organizations, particularly smaller ones with limited resources, this poses a substantial barrier to adopting AI-driven security measures. As a result, the benefits of AI in cybersecurity are not uniformly distributed, with larger, more resource-rich organizations being better positioned to leverage these advanced technologies.

Future Lines of Research

These limitations open up multiple avenues for future research, aiming to enhance the capabilities of AI in cybersecurity and address its current shortcomings.

Research into AI models that can effectively detect and defend against zero-day attacks is crucial. Such models would need to operate beyond the reliance on historical data, instead focusing on real-time analysis of network behavior, anomaly detection, and predictive algorithms that can anticipate unknown threats based on emerging patterns and indicators.

The development of synthetic data generation techniques represents a promising area of research. Synthetic data can serve as a stand-in for real-world data, enabling the training of AI models without the need for extensive datasets that may be difficult to obtain or raise privacy concerns. This approach could potentially address the data scarcity and privacy issues, providing AI models with the diverse and comprehensive data needed for effective learning and adaptation.

Future research in AI and cybersecurity must focus on detecting zero-day attacks in real-time, developing synthetic data for training, preventing AI misuse, democratizing AI for smaller organizations, and addressing ethical concerns. Interdisciplinary collaboration is crucial to unlock AI's full potential in cybersecurity, ensuring a safer digital landscape.

Addressing the potential misuse of AI in cyberattacks is another critical area of research. This involves not only the development of AI systems capable of identifying and neutralizing AI-powered threats but also the creation of ethical guidelines and security protocols to prevent the exploitation of AI technology for malicious purposes. This field would benefit from research that crosses disciplinary boundaries, incorporating insights from cybersecurity, AI ethics, law, and technology policy.

The exploration of cost-effective AI solutions in cybersecurity is essential, particularly for smaller organizations. Future research should focus on developing scalable and accessible AI tools that do not require extensive resources or expertise. This democratization of AI in cybersecurity would help level the playing field, allowing a broader range of organizations to protect themselves effectively against cyber threats.

Furthermore, the integration of AI into cybersecurity raises numerous ethical considerations. Future research should go deeper into the development of AI ethical frameworks that ensure transparency, accountability and fairness in AI-driven cybersecurity measures. Addressing these ethical dimensions is imperative to maintain public trust and ensure the responsible use of AI in a domain as critical as cybersecurity.

Finally, interdisciplinary research is vital in pushing the boundaries of AI applications in cybersecurity. Collaborations between computer scientists, cybersecurity experts, ethicists, legal scholars, and data scientists can lead to more holistic and robust AI solutions. Such collaborative efforts can address the multifaceted challenges of AI in cybersecurity, from technical hurdles to ethical dilemmas, paving the way for more advanced, ethical, and effective AI-driven cybersecurity practices.

AI has significantly enhanced the field of cybersecurity, its full potential is yet to be realized. The limitations present in current AI applications in cybersecurity underscore the need for continued research and innovation. By exploring these future research avenues, there is considerable potential to harness AI's capabilities more fully, advancing the field of cybersecurity to new heights and creating a safer digital environment.

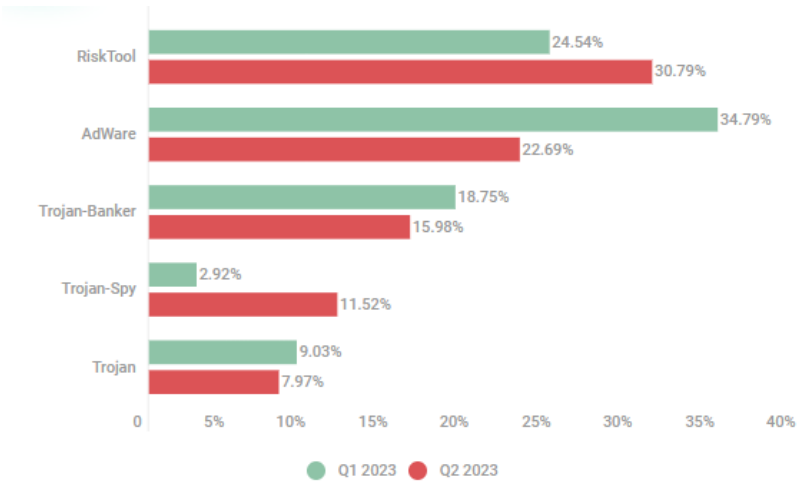
Malware Detection: Neural Insights

3

3.1 The Rise of Mobile Malware Threats

In the last few decades, the use of mobile devices has increased due to the large number of conveniences they offer. There is a wide variety of operating systems on the market, but among them there are two that stand out from the rest. Android and iOS account for 99.4% of the mobile OS market in 2022 [244]. Among them, Android leads with a 71.8% and IOS with 27.6%. Due to these figures, users of these operating systems have a greater choice of mobile applications as well as new functionalities on a regular basis. This aspect also implies a series of risks to be taken into account, as the increase in malware is closely linked to this growth.

Broadly speaking, we can define 4 categories of malware that have a higher incidence rate nowadays: Spyware, Adware, Ransomware and Banking Trojans Figure 3.1. Spyware collects and transmits personal data to attackers, including login credentials, contacts, and browsing history, and can activate a device's camera or microphone [219]. Adware displays unwanted ads and can slow down device performance, often bundled with free apps [81]. Ransomware encrypts data and demands payment for decryption, locking users out of their device [49]. Banking Trojans steal financial information, including credit card numbers and bank account information [45].



The widespread use of mobile devices, primarily Android and iOS, has brought convenience but also increased malware risks. Four major malware categories include Spyware, Adware, Ransomware, and Banking Trojans. Android's openness to third-party stores poses higher malware risks compared to iOS's restricted ecosystem. Official Android stores have some anti-malware measures.

Figure 3.1: In the first and second quarters of 2023, the distribution of newly identified mobile malware varied by type. The predominant category was unwanted software, specifically RiskTool. A considerable portion of these threats involved obfuscated Robtes files. Adware was prevalent, accounting for 22.69% of the threats. The most common adware families[141].

Openness of Android market poses higher malware risk than iOS due to third-party stores. Official Android store has anti-malware techniques, but unknown in third-party stores. iOS doesn't allow third-party stores except in the EU. Coexistence of app stores in Android doesn't have a big impact on risk.

3.2 Evolution of Malware and Countermeasures

The evolution of malware in mobile devices has followed a trajectory of increasing sophistication and variety, adapting to changing technologies and user behaviors. In recent years, there has been a significant increase in the prevalence and complexity of mobile malware.

One of the main methods of spreading mobile malware is through malicious applications and downloads. These infected applications often disguise themselves as legitimate software, making it difficult for users to distinguish between genuine and malicious ones. In addition, phishing attacks and social engineering techniques are also prevalent in the mobile ecosystem, deceiving users into revealing sensitive information or downloading malicious applications [241].

Mobile malware has significantly evolved, becoming more sophisticated and varied, adapting to technology and user behavior changes. Malicious apps, phishing, and social engineering are key spread methods. Android is particularly vulnerable, with Trojans like Xenomorph and Anatsa targeting financial data. The landscape also includes mobile ransomware, advertising Trojans, and advanced threats like Skygofree. Defenders employ various analysis and detection techniques, including static, dynamic, and hybrid analysis, as well as signature-based and anomaly-based detection, in this evolving cybersecurity battleground.

Regarding operating systems, Android is particularly susceptible to malware attacks due to its open nature, especially when users download applications from unofficial sources. Malware was found on 1 out of every 20 Android devices in 2022. Recent examples of mobile banking malware on Android include the Xenomorph and Anatsa Trojans, which focus on stealing login credentials, credit card information, and other financial data through overlay attacks and keylogging.

Moreover, mobile ransomware remains a significant threat to both consumers and businesses, with techniques and capabilities that have remained almost unchanged. Mobile advertising Trojans, once the top mobile malware threat, have had to change their techniques due to the decrease in the number of devices running older versions of Android, which are the main targets of these Trojans.

Mobile malware is not just an opportunistic tactic for cybercriminals; it is also being used as part of targeted, prolonged

campaigns that can affect many victims. A notable example is Skygofree[1], an advanced mobile implant used for targeted cyber-surveillance, active since 2014 [250].

1: Skygofree is an advanced mobile spyware capable of intercepting communications and monitoring user activities.

The landscape of malware spread and detection techniques is a constantly evolving domain, marked by the ingenuity of attackers and the persistence of defenders. Malware, in its quest to infiltrate and propagate, employs a variety of sophisticated methods[178]. Rootkits, for instance, are software tools that facilitate unauthorized, privileged access to a system, allowing remote control and further attacks[215]. Notorious examples like DroidDream have demonstrated their potency by injecting malicious code into apps on the Google Playstore[127].

Keyloggers[2] and spyware represent another insidious threat, capturing sensitive information such as banking credentials and personal data. Mysterybot, a combined keylogger, ransomware, and banking trojan, exemplifies the multifaceted nature of modern malware threats. Similarly, spyware like Flexispy invades privacy by intercepting communications and monitoring social media activity [211].

2: Keyloggers are types of software designed to record keystrokes made by a user on a device. They silently operate in the background, capturing every keystroke, including passwords, messages, and other sensitive information.

Adware, while seemingly benign, also poses a significant threat by illicitly collecting and transmitting user data for marketing purposes. This is often achieved through displaying targeted ads or redirecting browsers to sales websites. Scareware, like AndroidDefender, manipulates users into purchasing unnecessary software by exploiting fear through fake alerts.

Ransomware, a more direct form of cyber extortion, encrypts files or locks systems, demanding a ransom for release. The evolution of ransomware has led to sophisticated variants like Filecoder, which spreads through online forums and encrypts device contents.

Moreover, the emergence of crypto-mining[3] malware, or cryptojacking, highlights the shift towards exploiting computing resources for cryptocurrency mining without damaging or stealing data. This form of malware, exemplified by the draining of CPU processing power, underscores the varied objectives of modern cybercriminals.

3: Cryptocurrency mining, is the process of using computer resources to validate transactions and secure a blockchain network, earning cryptocurrency rewards in return.

Botnets, networks of infected devices controlled by a single command server, utilize compromised machines for large-scale attacks like DDoS, spam distribution, and click fraud

(Table 3.1). The evolution of botnets and the advent of fileless malware, which operates in main memory to avoid detection, signal a move towards more stealthy and resilient forms of cyber threats.

Table 3.1: Top 5 countries with IP address locations of servers used to control computers infected with malware [239].

Country	Nº Infected IPs
China	681867
USA	428741
India	267031
Indonesia	157051
Algeria	127940

On the defense front, anti-malware engines employ a combination of analysis and detection methods. Malware analysis includes static, dynamic, and hybrid approaches. Static analysis examines code without execution, offering quick and extensive coverage but struggling against obfuscation techniques. Dynamic analysis counters this by executing code in controlled environments, observing behavioral parameters like API calls and memory writes. Hybrid methods merge these approaches, leveraging their combined strengths [229].

For malware detection, signature-based and anomaly-based methods are predominant. Signature-based detection relies on a database of known malware signatures, necessitating constant updates to remain effective against new threats. Anomaly-based detection, alternatively, identifies deviations from normal patterns, flagging them as malicious. This method, while adept at recognizing unknown malware and zero-day attacks, can suffer from false positives.

The interplay between the cunning of malware and the sophistication of detection techniques represents a dynamic battleground in cybersecurity. As malware becomes more advanced, so too must the methods to detect and neutralize it, underscoring the ongoing arms race in the digital realm.

3.3 Leveraging Deep Learning for Malware Detection

The architecture of Android malware detection using deep learning involves several key stages: static and dynamic analysis of the Android APK file using tools like Androguard, APKTool, and DroidBox; extraction of features like API call sequences and system call features; and conversion of these features into a format suitable for deep learning analysis, such as RGB images or numerical vectors [286].

In the feature extraction phase, tools like IDA Pro and APKtool decompile APK files to extract static features from

various components of the Android application. Dynamic analysis is then performed using platforms like DroidBox, which installs applications in an Android Virtual Device and extracts dynamic features from the running log.

The extracted features undergo a selection process to eliminate noise, irrelevancy, and redundancy. This is accomplished using algorithms like association rule mining, information gain, chi-square statistical analysis, and frequency sorting. The features are then vectorized using methods like Word2Vec, one-hot encoding, and Euclidean distance measurement. These vectors serve as input for the neural network model [268].

The Android malware detection architecture using deep learning includes analyzing APK files, extracting features like API calls, and converting them into deep learning-compatible formats. Feature extraction involves tools like IDA Pro and APKtool, with noise reduction through algorithms like information gain. The selected features are vectorized for neural network analysis. The system's effectiveness is measured using metrics like True Positive Rate and Precision, but faces challenges like concept drift and adversarial attacks, necessitating continual model and dataset updates and new method development.

Evaluation metrics for the system include True Positive Rate (TPR), False Negative Rate (FNR), False Positive Rate (FPR), True Negative Rate (TNR), Accuracy, Precision, and F-measure. These metrics provide a comprehensive understanding of the system's ability to correctly identify malware and benign samples.

However, deep learning-based Android malware detection systems face several challenges. One significant issue is the frequent alteration of malware to evade detection, leading to concept drift in deep learning models. This drift results in the degradation of the model's detection effectiveness over time, necessitating regular updates to models and datasets.

Another challenge is adversarial attacks, where attackers manipulate inputs to cause misclassification by the model. This can occur through evasion attacks, where malicious samples are altered during testing to be classified as benign, and poisoning attacks, where the training data is contaminated to reduce the model's accuracy.

To combat these challenges, researchers continue to refine and develop new methods, such as using transfer learning and exploring novel neural network architectures[4]. The ongoing evolution and improvement of deep learning techniques are essential for maintaining the effectiveness of Android malware detection systems in the face of ever-changing malware threats and tactics.

4: Transfer learning allows leveraging existing models for new problems, enhancing learning efficiency and accuracy, while novel architectures provide more robust and adaptable solutions to the challenges presented by advanced malware.

3.4 Our Neural Approach to Malware Classification

In this field in particular, we have carried out work to deepen and define new methods to address all these aspects. Taking into account recent work and advances, we have studied the application of convolutional neural networks (CNN) for malware detection. As we have already discussed in previous sections, there is a considerable increase in studies that address malware detection using artificial intelligence, and after observing in other fields of science the positive results that have been obtained by applying CNNs, we have decided to adapt and work on new methods for our task.

Throughout this section, we will detail the objective, how we have approached the task and the results we have obtained.

3.4.1 Android Bytecode as a Neural Input

To be able to use Android applications as input for our model, we need to understand what CNNs are and how they work. CNNs are a class of deep neural networks highly effective in analyzing visual imagery. They are particularly well-suited for image recognition and classification tasks due to their unique architecture[5], which mimics the way the human brain processes visual information.

5: CNN architecture typically includes convolutional layers, pooling layers, and fully connected layers for feature extraction and classification.

The core concept of CNNs lies in their use of convolutional layers, which apply a series of learnable filters to the input image. These filters are small in size but extend through the full depth of the input volume. As the filters slide over the image, they perform element-wise multiplication with the part of the image they are covering, creating a feature map that encodes spatial hierarchies of features. This operation allows CNNs to capture the spatial and temporal dependencies in an image through the application of relevant filters, making them adept at recognizing patterns like edges, textures, and shapes in the input image.

The reason CNNs use images as input is that images are composed of pixel values, which can effectively represent various features and objects. By processing these pixel values, CNNs can detect patterns that are crucial for image recognition tasks. Each convolutional layer within the network extracts a

specific set of features; early layers may detect simple features like edges and curves, while deeper layers can identify more complex features like faces or objects.

Moreover, CNNs employ pooling layers to reduce the spatial size of the representation, thus decreasing the number of parameters and computations in the network. This dimensionality reduction helps in controlling overfitting.

The architecture of CNNs makes them particularly suitable for tasks where the context and locality of pixel values are important, which is a common scenario in image processing. This is why CNNs have become the model of choice for computer vision tasks, ranging from image and video recognition to medical image analysis. Their ability to learn hierarchical patterns in data makes them exceptionally powerful for these applications.

CNNs utilize convolutional layers with learnable filters to process images, capturing spatial and temporal dependencies by recognizing patterns like edges and shapes. They reduce dimensionality through pooling layers, controlling overfitting, making them ideal for computer vision tasks by learning hierarchical data patterns. This necessitated transforming Android applications into images for analysis in the experiment.

This is why for our experiment we had to work first on developing a methodology for transforming android applications into images so that they can be analysed by our model.

3.5 Image Representation of Android Bytecode

The source code of Android applications is not commonly available, so a common practice is to analyse the bytecode[6] of the app. The bytecode is contained in a Dalvik executable file with a "dex" extension. The "dex" (Dalvik Executable) files of Android apps contain compiled code that is executed by the Dalvik virtual machine, which is the runtime environment used by Android. The "dex" file format is optimized for small size and efficient execution on mobile devices with limited resources. This file contains all the information about classes, methods, strings, etc. maintaining always the same structure. "dex" files are compressed together with other relevant files such as resources, a folder with compiled code, libraries, etc.

6: Android bytecode is a set of instructions compiled from Android app code, executed by the Dalvik or Android Runtime.

Our method is based on the graphical representation of all the information in the application. To do this, the "dex" file is extracted from the application. We perform a dump of the binary file, carrying out a conversion of the entire byte stream to decimal. The values obtained are between 0 and

255. Each value obtained has been used to generate an RGB representation, the value being a specific RGB channel. There are a multitude of possible graphical representations. Studies have shown that CNNs perform worse on grayscale images, so our method proposes to generate graphical representations in other scales. Our methodology is shown in Figure 3.2.

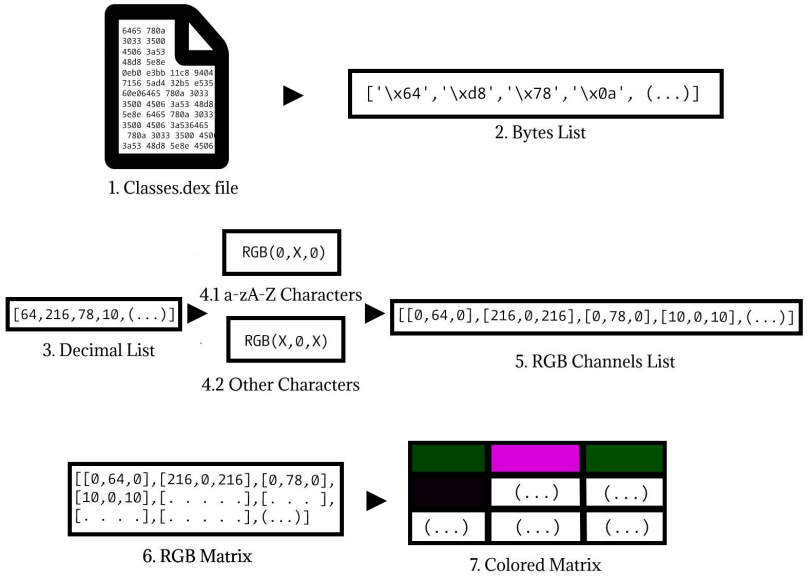


Figure 3.2: The data transformation consists of 5 main steps in which Android applications are transformed from the .APK file to a graphical representation without losing any data.

The "dex" files are structured sequentially with the different components of the application, starting with the header, followed by the IDs, strings, etc. Depending on the size of the application, in the graphical representations, the components will be placed at different heights. That is why, in order to give more relevance to the strings (ASCII values between 31 and 127), they have been represented in another RGB channel different from the rest of the values.

Our approach involves dividing the entire data stream into x subsequences, where x is determined by the pseudo formula $x = ceil(\sqrt{y})$ and y represents the length of the data stream. The function $ceil()$, short for "ceiling", is a mathematical function that rounds a given number up to the nearest integer greater than or equal to that number. Our goal is to generate a matrix, each subsequence is treated as a row, resulting in a matrix representation of the "dex" file that is $x * x$ in size. To account for subsequences that are shorter than x, we use the Zero Padding technique [109] at the end of those subsequences. All these elements in the matrix represent one byte of the "dex" file by means of a list of 3 values corresponding to the RGB channels. The last step is the

generation of the image, for which each element of the matrix is transformed into a pixel with the corresponding colour with the RGB values. The image representation comprising two distinct samples, one classified as malware and the other benign, is shown in Figure 3.3.

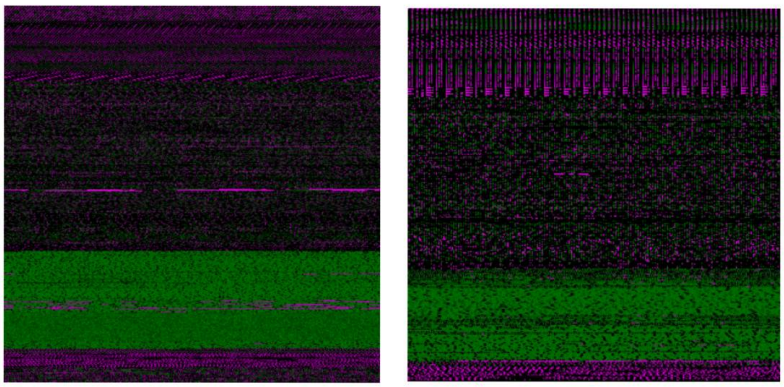


Figure 3.3: Two images generated from sample android applications are shown. On the left is a sample of benign code and on the right is a sample of an application with some malicious code.

3.5.1 Dataset and Preprocessing for Neural Training

Our research used a comprehensive dataset of Android APKs, named CICMalDroid. This dataset is a research initiative [172] that involves the dynamic analysis of Android samples using CopperDroid, a virtual machine introspection (VMI) based dynamic analysis system. The aim of this initiative is to automatically reconstruct low-level OS-specific and high-level Android-specific behaviors of the Android samples. In the course of this endeavor, a total of 17,341 samples were collected and analyzed. Of these samples, 13,077 were successfully executed, while the remaining samples failed due to various errors such as time-out, invalid APK files, and memory allocation failures. The successful execution of the majority of the collected samples demonstrates the efficacy of the approach adopted by CICMalDroid and provides valuable insights into the behavior of Android malware.

The research utilized the CICMalDroid dataset, comprising 17,341 Android APK samples analyzed using CopperDroid for dynamic analysis, aiming to reconstruct Android behaviors. Of these, 13,077 samples were successfully executed. The study focused on benign and unified malware categories, using 70% of malware samples for training, 20% for testing, and 10% for validation.

The dataset contains all samples classified into the following categories: Adware, Banking malware, SMS malware, Riskware, and Benign. Due to the small number of samples in some of the categories, it has been decided to work initially on two main categories, benign and, in a unified way, malware. Thus, our dataset consists of 5192 benign samples and 7581 malware samples. We used 70% for training, 20% for testing, and 10% for the model validation process.

3.5.2 Convolutional Neural Networks for Image-based Detection

The application of deep learning models, especially Convolutional Neural Networks (CNNs), has shown promising results in tackling this challenge. Among the myriad of models available, three have been distinguished for their effectiveness and suitability for image-based classification tasks: VGG16, ResNet50, and InceptionV3. The selection of these models is not arbitrary; it is grounded in their unique architectural innovations, proven track record in image classification challenges, and their adaptability to the context of Android malware detection, where malware signatures are often hidden in the application's code and can be visualized as images through various techniques.

Deep learning models like VGG16, RESNET50, and InceptionV3, known for their complex architectures, are used for Android malware detection through image classification. VGG16, with 16 layers, excels in extracting multi-level image features. RESNET50, using residual connections, addresses performance degradation in deep networks. InceptionV3's efficient architecture combines various filters for capturing features at different scales, employing regularization to prevent overfitting. These models are effective for pattern detection due to their innovative designs, feature extraction capabilities, and generalization.

VGG16 stands out for its simplicity and depth. Developed by the Visual Graphics Group at Oxford, which lends it the acronym VGG, this model is characterized by its 16-layer deep architecture. The genius of VGG16 lies in its uniformity, it exclusively uses 3 x 3 convolutional layers stacked on top of each other in increasing depth. By doing this, VGG16 can capture complex features from images, layer by layer, from basic textures and patterns in the initial layers to highly complex object features in the deeper layers. This depth, while increasing computational demand, enables the model to perform exceptionally well in image recognition tasks, making it an excellent choice for analyzing Android malware, where subtle and complex patterns must be discerned.

ResNet50, or Residual Network with 50 layers, tackles one of the most perplexing problems in deep learning: the vanishing gradient problem. As networks grow deeper, training them becomes increasingly difficult due to the degradation problem, where the accuracy starts saturating and then degrades rapidly. ResNet50 introduces an innovative solution with its use of residual blocks. These blocks allow the network to skip over certain layers, using identity shortcut connections that add the output from an earlier layer to a later layer, facilitating the training of much deeper networks. This breakthrough allows ResNet50 to learn features at multiple levels of abstraction, making it particularly adept at identifying the nuanced characteristics of malware in applications.

InceptionV3 represents a leap forward in optimizing network computation without sacrificing depth or width. It incorpo-

rates Inception modules, which parallelize convolutional layers with varying filter sizes, allowing the model to capture information at various scales efficiently. Furthermore, it introduces innovations such as factorization into smaller convolutions and the use of auxiliary classifiers to propagate label information deeper into the network. These innovations reduce the computational cost while maintaining high performance. Its efficient use of computational resources and ability to capture multi-scale features make InceptionV3 particularly well-suited for the constraints of malware detection, where diverse and subtle features must be detected efficiently.

The adoption of VGG16, ResNet50, and InceptionV3 in the domain of Android malware detection are the best options for pattern detection in images due to their innovative and computationally efficient architectures, their ability to efficiently capture features of different scales and levels of abstraction, their generalization capability to detect patterns in different types of images, their use of regularization techniques to prevent overfitting, and their availability as pre-trained models. Each model brings a unique set of strengths to the table: VGG16's deep and sequential architecture for capturing intricate details, ResNet50's ability to train very deep networks without performance degradation, and InceptionV3's computational efficiency and adaptability to varying feature sizes. Together, they form a robust foundation for developing high-performing, scalable, and efficient malware detection systems, capable of addressing the ever-evolving challenges in cybersecurity.

3.6 Experimental Insights and Model Evaluations

In this study, we explored the use of transfer learning to fine-tune pre-trained neural networks for our specific task. We conducted three separate experiments, each utilizing a different pre-trained network and training process.

Firstly, we employed the VGG16 network and trained it over 15 epochs, with 77 iterations each, using the SGD optimizer and a learning rate of 0.001. The result was an impressive accuracy of 98.05% during the training process.

Secondly, we utilized the RESNET50 network and carried out a transfer learning for our task, training it for 15 epochs with 69 iterations each, using the Adam optimizer and a learning rate of 0.01. The accuracy achieved during this training process was 97.34%.

The study explores transfer learning in fine-tuning pre-trained neural networks for a specific task, conducting experiments with VGG16, RESNET50, and InceptionV3 networks. High accuracies were achieved (98.05%, 97.34%, 97.48%). Precision, Recall, and F1 Score metrics are introduced to assess model performance, especially when balancing false positives and false negatives is crucial.

Lastly, we conducted an experiment with the InceptionV3 pre-trained network and fine-tuned it using transfer learning. The training process was carried out for 10 epochs, with 69 iterations each, using the SGD optimizer and a learning rate of 0.001. The resulting accuracy achieved during this training process was 97.48%.

Overall, these experiments demonstrate the effectiveness of transfer learning and the importance of choosing an appropriate pre-trained network and training process for a specific task.

To assess the performance of the proposed method, we define the use of the following metrics: Precision, Recall, and F1 Score.

Precision is defined as the number of true positives divided by the sum of true positives and false positives. It measures the proportion of positive predictions that are actually correct. This metric is important when the cost of a false positive is high, as it ensures that the model is correctly identifying the positive cases. Recall, on the other hand, is defined as the number of true positives divided by the sum of true positives and false negatives. It measures the proportion of actual positive cases that are correctly identified by the model. This metric is important when the cost of a false negative is high, as it ensures that the model is correctly identifying all positive cases. The F1 Score is a harmonic mean of precision and recall, emphasizing their balance. Precision measures correctness of positive predictions, while recall assesses coverage of actual positives. The F1 Score reaches its best value at 1 (perfect precision and recall) and worst at 0, serving as a comprehensive accuracy metric. It is calculated as shown in Equation 3.1. This metric is useful when precision and recall are both important and need to be balanced.

$$F1s. = \frac{2 * (precision * recall)}{(precision + recall)} \tag{3.1}$$

3.6.1 Performance Metrics and Outcome Analysis

Once we completed the training process of the three models, we carried out the validation process. During this process, we fed the models with a new set of data and obtained the prediction results. We then used these results to extract the prediction correlation matrix for each model. The correlation matrix helps to evaluate the accuracy and consistency of the model's predictions across the validation data set. The correlation matrix for the three models is shown in Figure 3.4.

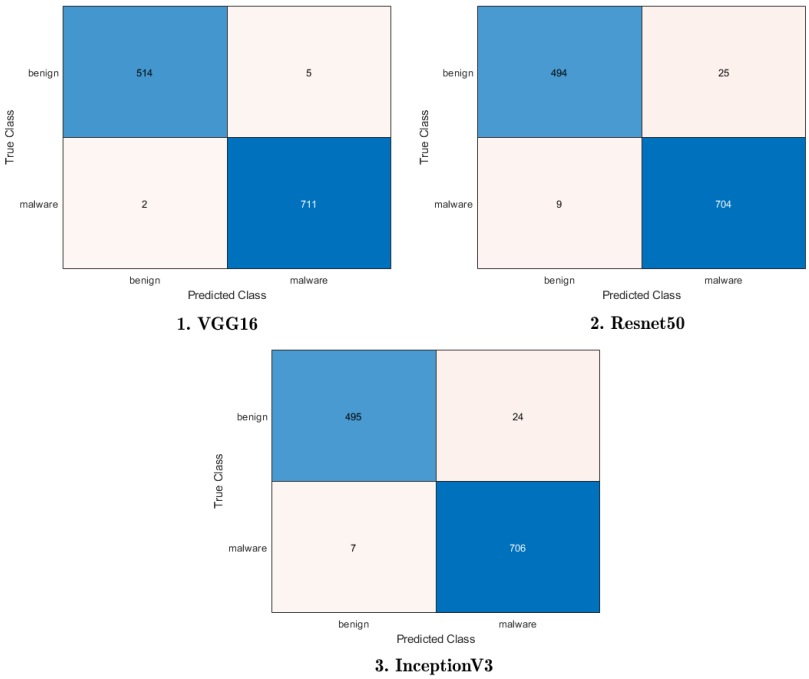


Figure 3.4: Classification confusion matrix of the 3 different models used, to assess performance.

Additionally, to further assess the performance of the models as mentioned in the previous section, we used Recall, Precision, and F1 Score as metrics. By using these metrics, we can gain a deeper understanding of the model's strengths and weaknesses and identify which model performs best on the validation data set. Table 3.2 shows the values for each of the models utilized.

Based on the F1 Score, precision, and recall metrics obtained for the three models, it appears that VGG16 has performed the best overall. It achieved the highest F1 Score of 0.995101, indicating that it has a good balance between precision and recall.

VGG16 had the highest recall score of 0.997195, indicating that it correctly classified a higher percentage of actual positives than the other two models. Resnet50 had the lowest recall score of 0.987377, indicating that it had a higher false negative rate compared to the other models.

Table 3.2: Assessing VGG16, Resnet50 and InceptionV3 Performance with Recall, Precision, and F1 Score Metrics

	Recall	Precision	F1 score
VGG16	0.997195	0.993017	0.995101
Resnet50	0.987377	0.965706	0.976422
InceptionV3	0.990182	0.967123	0.978517

On the other hand, InceptionV3 had quite similar results to Resnet50, indicating that it had a high rate of false positives compared to VGG16. When comparing RESNET50 and InceptionV3, we observed a slight improvement in the false negative rate, as InceptionV3 displayed better recall results. While it is challenging to determine which is more crucial in our case, false positives or false negatives, it is generally preferable to prevent the use of an application due to a false positive rather than risking device infection due to a false negative. Thus, we highlight InceptionV3 over RESNET50 due to its superior performance in recall, albeit a minor improvement.

Overall, it is important to consider both precision and recall when evaluating a model's performance. The F1 Score takes into account both of these metrics, providing a more comprehensive evaluation of the model's performance.

3.7 Conclusions and Pathways for Future Research

In this contribution, we present the performance of three specific Convolutional Neural Networks (CNN) models, such as VGG16, RESNET50, and InceptionV3, as detection models for Android malware. Our evaluation, which was conducted on a dataset of 13,000 applications, demonstrates that VGG16 model has the best-performing score up to 99% based on the F1 score, precision, and recall metrics. Another notable strength of VGG16 is its recall score, which indicates a low

false negative rate of malicious app detection. The effectiveness of these models further confirms the value of Android's graphical representation of bytecodes, emphasizing that the conversion process from DEX files to images is executed without any loss of information.

In conclusion, our research proved the viability of using Convolutional Neural Networks (CNN), specifically VGG16, RESNET50, and InceptionV3, to detect malware on Android by classifying images containing transformed bytecode sequences from the DEX file of the applications. Through the use of transfer learning and fine-tuning, we were able to achieve high accuracy rates in detecting malware samples, demonstrating the potential for this methodology to be applied in real-world scenarios. Finally, we hope our findings contribute to the ongoing efforts to improve malware detection and prevention for the Android ecosystem.

This study evaluates the performance of VGG16, RESNET50, and InceptionV3 Convolutional Neural Networks for Android malware detection. VGG16 achieves the highest F1 score (99%) and recall, showing potential for real-world applications. Future research will focus on explainability and detecting malware subcategories.

In future work, we plan to explore the explainability of these networks to better understand their decision-making processes. This will allow us to classify and understand new malware mutations and address the issue of code obfuscation. Furthermore, we aim to test its effectiveness in detecting the various subcategories of malware, which could represent an advancement in the ability to classify and categorize malware samples.

3.7.1 Improving Explainability in Neural Malware Detection

As I explained in the previous section, explainability, referring to the ability to understand and explain how and why an artificial intelligence model makes certain decisions or classifications, is one of the most relevant points to be improved in this research. Explainability is fundamental for the adoption and acceptance of this type of methodology. It would allow us to better understand false positives and false negatives, and adjust the model accordingly.

Enhancing explainability is a key focus of this research. Understanding AI model decisions is crucial for adoption. Techniques like visualizing model-identified features and using interpretable methods (e.g., LIME, SHAP) can help. Balancing accuracy and transparency is a priority for successful malware detection models in security.

Firstly, improving explainability may involve developing techniques that allow us to visualise the features that the network identifies as indicative of malware. This would not only help us validate and improve models, but also facilitate the communication of model decisions, thereby increasing confidence in this methodology.

One strategy to improve explainability could be the integration of interpretable machine learning techniques, such as LIME (Local Interpretable Model-agnostic Explanations)[287] or SHAP (SHapley Additive exPlanations)[14]. These techniques can break down the decisions of a complex model into understandable contributions of each feature, thus providing a detailed explanation of why an application was classified as malware.

It is also very important to consider that improved explainability should not compromise the effectiveness of the model. Therefore, a balanced approach that combines high malware detection performance with clear and meaningful explanations will be essential for the success of these models in critical security environments.This balance between accuracy and transparency is emerging as one of the main lines of future research in the field of malware detection using artificial intelligence techniques.

3.7.2 Extending Detection to Diverse Malware Types

Expanding CNNs' malware detection capabilities to cover diverse types is a vital research direction. Adapting CNNs to detect varied Android malware requires an enriched dataset and sensitivity to differentiated attack behaviors. Continuous model adaptation against evasion tactics is essential. Collaboration with cybersecurity experts enriches training and validation processes, making models more robust and knowledge-driven. This research aligns with strategic anticipation of future cybersecurity challenges.

Expanding the malware detection capability of Convolutional Neural Networks to encompass a wider variety of malware types is a significant challenge and an interesting direction for future research. This approach not only increases the utility of CNN models in environments other than cybersecurity, but also strengthens the overall robustness of the system against emerging and evolving threats.

In the context of malware in Android apps, malware types can vary considerably in their attack methods and behaviours, posing unique challenges for detection. Adapting CNNs to identify and differentiate between these various types of malware involves not only expanding the training database with varied examples, but also refining the models to be sensitive to the subtleties and specificities of each malware type. This could include incorporating more granular and specific features of application code that can indicate the presence of varied malicious behaviour.

Another very important aspect is the continuous adaptation of the model in the face of advanced evasion techniques employed by modern malware. As malware developers work

on strategies to avoid detection, CNN models must be updated and trained to recognise and respond to these evasive tactics. This could involve the use of continuous or adaptive learning techniques, where the model is regularly updated with information about the latest threats.

In addition, collaborating with cybersecurity experts and incorporating their specialised knowledge can enrich the process of training and validating CNNs. This can help ensure that models are not only data-driven, but also based on a deep understanding of the nature and tactics of malware.

Diversifying malware detection through CNNs can break new ground in Android app security, providing an additional layer of defence against a wider range of digital threats. This comprehensive, multi-faceted approach to improving malware detection represents a significant step forward in protecting devices and data in an increasingly digitised and connected world. Research in this area is not only technical but also strategic, focusing on anticipating and countering future malware evolutions in the cybersecurity landscape. Therefore, this research opens the door to try to address this aspect, as it could lead to new developments not only in the field of mobile cybersecurity but also in malware detection on any platform.

NetFlow Defense: CNN Surveillance

4

4.1 Overview of Network Security

Network security refers to a set of policies, practices, and tools designed to protect the integrity, confidentiality, and accessibility of computer networks and data. It involves a multitude of technologies, devices, and processes. In its simplest term, it is a set of rules and configurations designed to protect the integrity, confidentiality, and accessibility of computer networks and data using both software and hardware technologies. Every organization, regardless of size, industry, or infrastructure, requires a degree of network security solutions in place to protect it from the ever-growing landscape of cyber threats in the wild today.

Network security involves policies, practices, and tools to protect network integrity, confidentiality, and accessibility, essential for all organizations. It includes advanced technologies like firewalls, intrusion detection and prevention systems, and encryption protocols. The rise of cloud computing and IoT expands its scope, introducing new challenges in data security, user access management, and IoT device vulnerabilities.

The landscape of network security today is marked by a complex interplay of advanced technologies, evolving threats, and stringent regulatory requirements. The current state of network security is defined by a few key characteristics:

Modern network security is a sophisticated realm, leveraging cutting-edge technologies to protect against a myriad of cyber threats. Firewalls have evolved from simple packet filters to next-generation firewalls (NGFWs[1]) that integrate intrusion prevention, application awareness, and cloud-based threat intelligence. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) have become more advanced, using sophisticated algorithms to detect and respond to threats in real time.

1: NGFWs are advanced network security appliances that integrate traditional firewall features with advanced threat detection and prevention capabilities.

Encryption has become a cornerstone of network security, with protocols like Secure Socket Layer (SSL) and its successor, Transport Layer Security (TLS), being widely used to secure internet communications. Additionally, the adoption of cloud-based security services, known as Security as a Service (SECaaS[2]), is on the rise. These services provide scalable security solutions to organizations of all sizes, enabling them to manage a wide range of security functions including anti-virus, anti-malware, and firewall management remotely.

2: SECaaS, or Security as a Service, is a cloud-based security solution that delivers various security services and protections to organizations.

The proliferation of cloud computing and the Internet of Things (IoT) has expanded the scope and complexity of network security. Cloud computing presents new challenges in securing data and managing user access. Cloud security requires a different approach, focusing on perimeter security, internal threats, and inter-cloud traffic. The shared responsibility model in cloud security, wherein security obligations are divided between the cloud provider and the client, has become a standard practice.

Similarly, IoT devices, which often lack built-in security features, present unique vulnerabilities. The interconnected nature of these devices can potentially open up new avenues for cyberattacks, making the network security of IoT devices a critical concern. Security strategies for IoT focus on endpoint security, secure communications, and regular updates to address vulnerabilities.

4.1.1 Importance of Network Security

The importance of network security cannot be overstated, and its role is crucial in the modern digital world. As organizations and individuals increasingly rely on the internet and digital networks for daily operations, the potential impact of a security breach has also escalated. Network security serves several essential functions:

Network security is paramount in the digital age. It safeguards sensitive data, defends against cyber threats, ensures regulatory compliance, maintains privacy, enables secure communication, and fosters trust. Vital for organizations, it protects reputation, legal standing, and financial health, making it a cornerstone of modern strategy.

- Protects Sensitive Data: One of the most valuable assets of any organization is its data. Network security ensures that this data is kept safe from unauthorized access, breaches, and theft. This is particularly crucial for data that falls under privacy laws and regulations, like personal data, medical records, financial data, etc.
- Defends Against Cyber Threats: The digital landscape is riddled with various forms of cyber threats such as viruses, worms, spyware, and ransomware. Network security helps to identify and block these threats before they infiltrate the network.
- Ensures Regulatory Compliance: Many industries are governed by regulatory requirements that dictate how data must be protected. Network security helps organizations comply with these regulations, avoiding legal consequences and fines.

- Maintains Privacy: For both individuals and organizations, network security is fundamental to maintaining privacy. It prevents the unauthorized sharing of personal information and protects against identity theft.
- Supports Reliable and Secure Communications: Effective network security enables safe and reliable communication between users and systems, ensuring that communications are not intercepted or tampered with.
- Enables a Stable and Efficient Network: By protecting against disruptions caused by network attacks, security measures ensure that the network remains stable and efficient, maintaining uptime and productivity.
- Cultivates Consumer Trust: Organizations that demonstrate a commitment to network security can build and maintain trust with their clients and partners, as they are seen as responsible custodians of data.

Network security, therefore, is not just a technical requirement but a critical component for any organization's overall strategy. It safeguards an organization's reputation, legal responsibilities, and financial well-being.

4.1.2 Early Days of Network Security

The roots of network security can be traced back to the early days of computer networking. Initially, security wasn't a major concern because networks were more isolated and the number of users was limited.

Network security traces its origins to early computer networking, where security wasn't a priority due to isolated and limited networks. With the birth of ARPANET and the emergence of computer viruses in the 1980s, the need for network security became evident, leading to its development and evolution.

- Birth of ARPANET: In the late 1960s, the Advanced Research Projects Agency Network (ARPANET) was developed. This was the first network to implement the TCP/IP protocol suite, which would become the foundation for today's internet. Security in these early networks was minimal, largely because they were closed systems used by a small, trusted community.
- The Advent of Viruses and the Need for Security: As personal computers became more popular in the 1980s, the first computer viruses emerged. This led to an increased awareness of the need for network security. The first known computer virus, 'Elk Cloner,' spread on Apple II systems in 1982, and the infamous 'Morris

worm' in 1988 brought the vulnerabilities of networked environments into sharp focus.

4.1.3 The Growth of the Internet

This period marked the rapid expansion of the internet, bringing with it new security challenges.

- Widespread Internet Adoption: The 1990s saw the commercialization and expansion of the internet. With more people and organizations going online, the potential for security breaches grew.
- Notable Cyber Attacks: The period was marked by several significant cyber attacks. The Melissa virus (1999), the ILOVEYOU worm (2000), and the Code Red worm (2001) were some of the notable examples that affected millions of computers worldwide, leading to a heightened awareness of network security.
- Development of Security Protocols and Tools: In response to these threats, the development of security protocols and tools gained momentum. Firewalls, antivirus software, and the concept of intrusion detection systems started to evolve during this time.

4.1.4 Era of Cybersecurity Awareness

The new millennium has seen a continued escalation in cyber threats, matched by a more sophisticated approach to network security.

- Advanced Cyber Threats: The sophistication and frequency of cyber attacks have increased dramatically. Advanced persistent threats (APTs)[3], state-sponsored hacking, and large-scale data breaches have become more common.
- Rise of Cybersecurity Regulations: In response to these threats, governments and international bodies have introduced various cybersecurity regulations. The General Data Protection Regulation (GDPR)[4] in the EU and other similar laws around the world have set new standards for data protection and privacy.

3: Advanced Persistent Threats (APTs) are highly sophisticated, stealthy, and prolonged cyberattacks orchestrated by skilled threat actors to infiltrate and compromise targeted systems.

4: GDPR is a data privacy regulation that imposes strict rules on personal data handling, storage, and protection.

- Integration of AI and Machine Learning: Recent years have seen the integration of artificial intelligence (AI) and machine learning into network security solutions. These technologies are being used for threat detection, pattern recognition, and automated response to security incidents.
- The Challenge of Emerging Technologies: The rise of cloud computing, the Internet of Things (IoT), and mobile computing has expanded the security perimeter, creating new challenges and complexities in network security.

4.2 Challenges in Network Security

Network security is an ever-evolving field that plays a pivotal role in safeguarding the integrity, confidentiality, and availability of digital information in our interconnected world. As technology continues to advance at an unprecedented pace, so do the challenges and complexities faced by organizations and individuals alike in securing their networks. This section dives into the dynamic landscape of network security, from the constant flow of cyber-attacks to the weaknesses of modern network designs. We will also examine the latest advances and trends in network security, including cutting-edge technologies, best practices, and evolving methodologies that help fortify networks against the ever-expanding threat landscape.

4.2.1 Common Types of Cyber Attacks in Network Security

In today's hyperconnected digital age, the importance of network security cannot be overstated. Organizations and individuals rely on networks to transmit, store, and access vast amounts of sensitive information. However, this very reliance on networks has made them prime targets for cybercriminals and malicious actors seeking to exploit vulnerabilities for their gain. Understanding the common types of cyber attacks in network security is paramount for building robust defenses and safeguarding sensitive data.

Malware Over Networks

Malware, short for malicious software, is a class of cyber threats that poses a significant risk to network security and data integrity. This section will explore the world of malware, how it infiltrates network systems and explore the far-reaching consequences of malware infections on network security and data integrity.

Malware, harmful software designed to compromise network security and data integrity, infiltrates systems via email attachments, infected websites, removable media, and software vulnerabilities. Its consequences include data theft, financial losses from ransomware, operational disruptions, reputational damage, and legal/regulatory repercussions.

Malware is a broad term that refers to any software that is specifically designed to harm, disrupt, or compromise computer systems, networks, or user data. It operates stealthily, often without the user's knowledge or consent, and its objectives range from stealing sensitive information to gaining control over a compromised system. Malware can infiltrate network systems through various vectors, including:

- Email Attachments: Malicious attachments in seemingly innocuous emails can be a common entry point for malware. When unsuspecting users open these attachments, malware can execute and infect the local system or network.
- Infected Websites: Visiting compromised or malicious websites can trigger drive-by downloads, where malware is automatically downloaded and installed on the user's device.
- Removable Media: Malware can spread through infected USB drives, external hard disks, or other removable media when plugged into a network-connected device.
- Software Vulnerabilities: Exploiting vulnerabilities in software, operating systems, or network devices is another method employed by malware to gain access to network systems.

The consequences of malware infections can be devastating for network security and data integrity. Some of the key impacts include:

- Data Theft: Malware can exfiltrate sensitive information, including personal data, financial records, and intellectual property, leading to significant privacy breaches.
- Financial Loss: Ransomware attacks can result in substantial financial losses, as organizations may need to pay ransoms or invest in recovery efforts.

- Operational Disruption: Malware-induced system failures or network outages can disrupt business operations, leading to downtime and productivity losses.
- Reputation Damage: Data breaches caused by malware can tarnish an organization's reputation and erode trust among customers and partners.
- Legal and Regulatory Consequences: Non-compliance with data protection regulations due to a malware breach can result in legal penalties and regulatory sanctions.

Man-in-the-Middle Attacks

Man-in-the-Middle (MitM) attacks represent a sophisticated and surreptitious class of cyber threats that undermine the integrity and confidentiality of network communications. In this section, we will unravel the concept of MitM attacks, elucidate how they work, and explore the multitude of methods attackers employ to intercept and manipulate network communications. Furthermore, we will underscore the grave risks associated with unauthorized access and data interception in such attacks.

Man-in-the-Middle (MitM) attacks, a sophisticated cyber threat, involve unauthorized interception and manipulation of communications between two parties. Attackers use methods like ARP and DNS spoofing, SSL stripping, and Wi-Fi eavesdropping, leading to severe consequences such as data breaches, identity theft, data tampering, privacy invasion, and financial losses.

At its core, a Man-in-the-Middle (MitM) attack is a type of cyberattack where an unauthorized entity secretly intercepts and relays communications between two parties without their knowledge or consent. In such attacks, the attacker positions themselves "in the middle" of the communication flow, masquerading as the legitimate sender or receiver (Figure 4.1), while eavesdropping on, altering, or even injecting malicious content into the conversation.

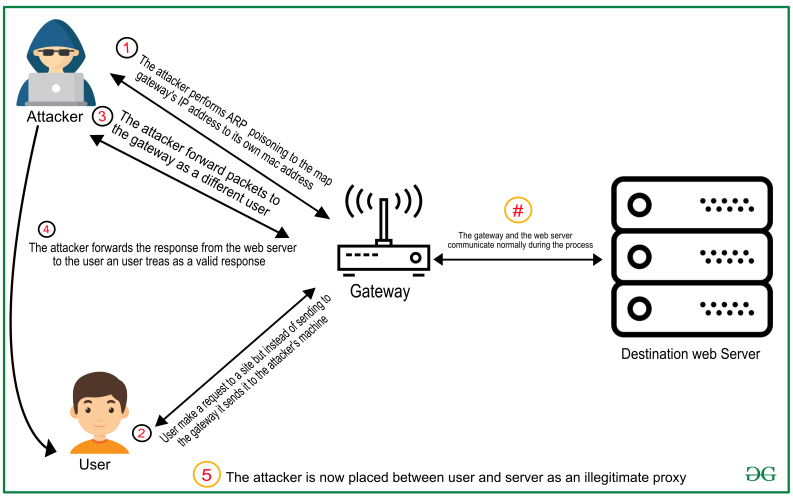


Figure 4.1: Shows how an ARP spoofing technique is used, enabling a man-in-the-middle attack to intercept and manipulate data [93].

MitM attacks exploit vulnerabilities in the communication channels between two parties. The attack typically unfolds as follows:

- Interception: The attacker secretly intercepts the communication flow between the victim and the intended recipient.
- Impersonation: The attacker may impersonate one or both parties, leading Alice to believe she is communicating with Bob, and vice versa.
- Data Manipulation: The attacker can alter the messages or data being exchanged, introducing errors, misinformation, or malicious content into the conversation.
- Data Theft: The attacker may also steal sensitive information, such as login credentials, financial data, or confidential documents, as it passes through their position.

Attackers utilize various methods to execute MitM attacks, including:

- ARP Spoofing: Address Resolution Protocol (ARP) spoofing involves manipulating ARP tables to link the attacker's MAC address with the IP address of a legitimate network device, redirecting traffic through the attacker's system.
- DNS Spoofing: Attackers can compromise Domain Name System (DNS)[5] servers to redirect users to malicious websites, intercepting their communications.
- SSL Stripping: In this technique, the attacker downgrades a secure HTTPS connection to HTTP, allowing them to eavesdrop on unencrypted traffic[6].
- Wi-Fi Eavesdropping: Attackers may set up rogue Wi-Fi access points to intercept traffic from unsuspecting users connected to their network.

5: DNS is a distributed system translating human-readable domain names to IP addresses for routing internet traffic efficiently.

6: HTTPS (Hypertext Transfer Protocol Secure) encrypts data between the user and the server, while HTTP (Hypertext Transfer Protocol) transmits data in plaintext.

The consequences of successful MitM attacks can be severe and encompass:

- Data Breach: Attackers can capture sensitive data, including usernames, passwords, credit card details, or confidential business information.
- Identity Theft: By impersonating victims, attackers can perform unauthorized actions on their behalf, such as conducting financial transactions or changing account passwords.

- Data Tampering: Manipulating data in transit can lead to erroneous or malicious changes in messages or files, compromising their integrity and accuracy.
- Privacy Invasion: MitM attacks violate individuals' privacy rights by intercepting and scrutinizing their private communications.
- Financial Loss: Fraudulent transactions and unauthorized access can result in significant financial losses for both individuals and organizations.

Man-in-the-Middle attacks pose a substantial threat to network security and data confidentiality. It is imperative for individuals and organizations to be aware of MitM attack vectors and employ robust security measures, such as encryption, secure communication protocols, and regular system monitoring, to mitigate the risks associated with these insidious cyber threats. Vigilance and proactive defense strategies are essential in safeguarding network communications against the clandestine menace of MitM attacks.

Denial of Service (DoS) Attacks

Denial of Service (DoS) attacks represent a menacing category of cyber threats designed to disrupt and incapacitate network services, rendering them inaccessible to legitimate users. It is a type of network attack that has seen a huge increase in recent years. Cisco's[7] prediction anticipates a global DDoS attack increase, from 7.9 million in 2018 to 15.4 million in 2023, marking an 807% rise in incidents over nine years Figure 4.2. Quarterly attacks grew from approximately 325,000 in Q1 2013 to around 2.9 million in Q1 2022 based on historical data and projections[56].

7: Cisco is a multinational technology company specializing in networking, hardware, software, and services, playing a significant role in IT infrastructure.

In this section, we will define DoS attacks and look into their objectives in order to disrupt network services.. We will also explore the various types of DoS attacks, including flooding attacks and resource exhaustion attacks, and elucidate the significant impact these attacks have on network availability and performance.

A Denial of Service (DoS) attack is a deliberate and malicious attempt to make a network service, website, or online resource unavailable to its intended users. The primary objective of a DoS attack is to overwhelm or incapacitate the target system's resources, rendering it incapable of processing legitimate

user requests. In essence, the attacker's goal is to disrupt normal network operations and compromise the availability of services.

DoS attacks come in various forms, but they can be broadly categorized into two main types:

DoS attacks, mainly flooding and resource exhaustion types, overwhelm targeted systems by inundating them with traffic or depleting resources. Common methods include UDP, SYN, ICMP floods, HTTP/HTTPS request floods, Slowloris, and DNS Amplification attacks. They cause service disruptions, downtime, revenue loss, operational inefficiency, and increased resource costs.

- Flooding Attacks: In flooding attacks, the attacker inundates the target system with an overwhelming volume of traffic or requests, causing it to become overwhelmed and unable to respond to legitimate requests. Common flooding attacks include:
  - UDP Flood: Attackers send a large number of User Datagram Protocol (UDP) packets to a target, often targeting vulnerable network infrastructure components like routers or DNS servers.
  - SYN Flood: SYN flood attacks exploit the TCP three-way handshake process, overwhelming the target server by initiating numerous incomplete connections.
  - ICMP Flood: Attackers flood the target with Internet Control Message Protocol (ICMP) Echo Request packets, often referred to as "ping" floods, to consume its resources.
- Resource Exhaustion Attacks: Resource exhaustion attacks focus on depleting the target system's essential resources, such as CPU, memory, or network bandwidth. These attacks include:
  - HTTP/HTTPS Request Flood: Attackers send a massive number of HTTP or HTTPS requests to a

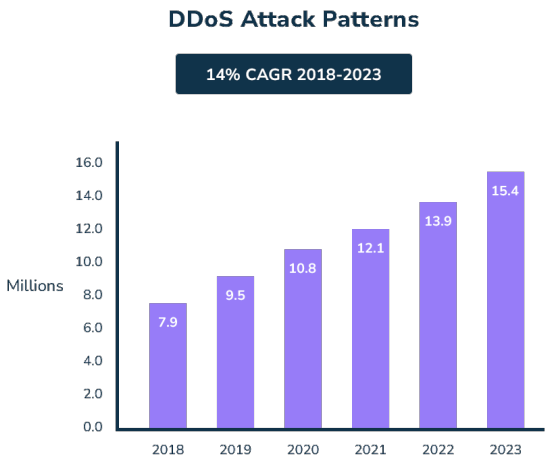


Figure 4.2: Compound Annual Growth Rate (CAGR) of Distributed Denial of Service (DDoS) attacks from 2018 to 2023, illustrating a significant increase in frequency over the five-year period, with a consistent year-over-year growth rate of 14% [243].

web server, consuming its processing power and bandwidth.
- Slowloris Attack: This attack exploits the number of concurrent connections a server can handle, holding them open indefinitely, ultimately exhausting the server's capacity.
- DNS Amplification: Attackers misuse vulnerable DNS servers to amplify their traffic, directing it toward the target and causing resource exhaustion.

DoS attacks can have severe repercussions on network availability and performance, including:

- Service Disruption: The most apparent impact is the complete or partial unavailability of network services, websites, or online resources during the attack.
- Downtime: Organizations may experience prolonged downtime, leading to financial losses, damage to reputation, and customer dissatisfaction.
- Loss of Revenue: E-commerce platforms and online businesses may suffer significant financial losses due to interrupted transactions and loss of customer trust.
- Operational Inefficiency: DoS attacks can disrupt critical business processes, impede employee productivity, and hinder normal operations.
- Resource Costs: Mitigating DoS attacks often requires investing in additional infrastructure, security measures, and personnel, incurring extra expenses.

Denial of Service (DoS) attacks are a persistent threat to network availability and performance. Understanding their objectives, types, and the potential impact is crucial for organizations and individuals seeking to defend against these disruptive attacks. Implementing proactive security measures, such as intrusion detection systems, content delivery networks (CDNs), and load balancing, can help mitigate the risks and ensure the resilience of network services in the face of DoS attacks.

Advanced Persistent Threats (APTs)

Advanced Persistent Threats (APTs) represent a highly sophisticated and covert category of cyberattacks that can compromise the security of organizations over an extended period. In this section, we will define APTs and explore their

characteristics as stealthy, long-term attacks. We will also discuss how APTs target organizations, often with specific objectives like data theft or espionage, and emphasize the critical importance of continuous monitoring and robust defense mechanisms against these persistent threats.

Advanced Persistent Threats (APTs) are sophisticated cyberattacks targeting specific organizations with objectives like data theft or espionage. These stealthy, long-term attacks require continuous monitoring and robust defense strategies. APTs are characterized by their stealth, persistence, and targeted approach, often remaining undetected for extended periods. They typically aim to steal valuable data, conduct espionage, maintain persistent network access, or infiltrate supply chains. Effective defense against APTs involves implementing advanced threat detection, enhancing network security, raising user awareness, developing incident response plans, and sharing threat intelligence within the industry.

Advanced Persistent Threats (APTs) are a class of cyberattacks characterized by their stealthy, persistent, and highly targeted nature. These attacks are not characterized by their speed or aggressiveness; rather, they are designed to operate silently over an extended period, often evading detection and maintaining a long-term presence within a target's network. Key characteristics of APTs include:

- Stealth: APTs are stealthy by nature, often using advanced techniques to remain hidden within a compromised network. Attackers take measures to avoid detection and disguise their activities.
- Persistence: APTs are not one-time attacks; they persistently target organizations, remaining active for weeks, months, or even years. Attackers maintain a foothold within the target environment.
- Targeted: APTs are specifically tailored to target particular organizations, industries, or entities. Attackers conduct extensive reconnaissance to understand the target's vulnerabilities and objectives.

APTs are typically executed with a strategic focus, often aiming to achieve specific objectives, such as:

- Data Theft: APTs often target valuable intellectual property, sensitive corporate data, or personally identifiable information (PII). The stolen data can be exploited for financial gain, industrial espionage, or political purposes.
- Espionage: Nation-state actors or cybercriminal groups may use APTs for espionage, infiltrating government agencies, defense contractors, or organizations with access to sensitive information [269].
- Persistent Access: APTs maintain unauthorized access to a network, allowing attackers to observe, gather information, and launch further attacks over an extended period.
- Supply Chain Attacks: APTs may infiltrate an organization's supply chain [8], compromising software updates,

8: SolarWinds Case is well-known attack in which the attackers injected a backdoor into a software update of SolarWinds, a popular networking tool used by many high profile companies and government agencies. The backdoor allowed attackers remote access to thousands of corporate and government servers. The global-scale attack led to many data breaches and security incidents.

hardware components, or third-party vendors to gain access.

Given the prolonged and stealthy nature of APTs, continuous monitoring and proactive defense measures are essential. Organizations should:

- Implement Threat Detection: Employ advanced threat detection solutions that can identify unusual behavior or anomalies in network traffic and system activities.
- Enhance Network Security: Strengthen network security by regularly updating software, patching vulnerabilities, and enforcing robust access controls.
- User Awareness: Train employees to recognize phishing attempts, social engineering tactics, and other methods used by APT actors.
- Incident Response Plans: Develop and regularly test incident response plans to minimize the impact of APTs when detected.
- Information Sharing: Collaborate with industry peers and share threat intelligence to stay informed about emerging APT campaigns.

Advanced Persistent Threats (APTs) represent a formidable and persistent challenge to organizations. Recognizing their stealthy nature, targeted approach, and specific objectives is crucial for organizations seeking to defend against them. Continuous monitoring, threat detection, and a proactive security posture are fundamental components of effective defense against APTs in today's evolving threat landscape.

4.2.2 Emerging Threats in Network Security

As the digital landscape continues to evolve, so do the threats that imperil network security. In this section, we will take a closer look at two prominent emerging threats in network security: AI-generated attacks and threats to IoT devices.

AI-Generated Attacks

Artificial Intelligence (AI) has brought both opportunities and challenges to the realm of cybersecurity. Attackers are increasingly leveraging AI to automate and enhance their malicious activities. AI-generated attacks are a manifestation

of this trend, wherein machine learning algorithms are used to craft more effective and adaptable attack strategies.

AI-Generated Attacks and IoT Device Threats are two of the new risks threatening the cyber security landscape. AI is being used by attackers to create sophisticated, adaptable malware that can evade traditional security measures. Meanwhile, the increasing use of IoT devices, often with weak security, poses risks like botnet enlistment and data breaches. Addressing these threats requires advanced security solutions, continuous adaptation, and enhanced IoT device protection through regular updates, strong authentication, and thoughtful design. Staying ahead of these emerging threats is crucial for safeguarding network infrastructure and data.

AI-generated malware is a growing concern. It has the ability to evolve rapidly, adapting to changes in network defenses and evading traditional security measures[201]. These AI-driven attacks can target vulnerabilities more efficiently, making them a formidable threat to network security.

Detecting and mitigating AI-generated threats is a complex task. Attackers can use AI to generate realistic phishing emails, polymorphic malware, or even craft convincing deepfake content. Traditional signature-based detection methods may struggle to keep pace with such dynamic threats. This necessitates the development of advanced AI-driven security solutions and continuous adaptation to evolving attack tactics.

Threats to IoT Devices

The proliferation of Internet of Things (IoT) devices has introduced new dimensions to network security. IoT devices often lack robust security features, making them vulnerable to exploitation. These devices include smart thermostats, cameras, sensors, and even industrial machinery connected to the internet.

When IoT devices are compromised, they can be enlisted into botnets, used for distributed denial of service (DDoS) attacks, or exploited to gain unauthorized access to a network. Data breaches through compromised IoT devices can expose sensitive information, and in some cases, even threaten physical safety if they control critical infrastructure.

Securing IoT devices is imperative to prevent network vulnerabilities. This entails regular software updates and patches, strong authentication mechanisms, and the implementation of network segmentation to isolate IoT devices from critical systems. Additionally, manufacturers must prioritize security in IoT device design and production.

4.2.3 Vulnerabilities in Modern Networks

Modern networks, with their intricate interplay of software, hardware, and human elements, are not immune to vulner-

abilities that threaten their security and integrity. In this section, we will examine the vulnerabilities that exist within modern networks, including weaknesses in software and hardware, the role of human factors, and the complexities introduced by modern network architectures.

Weaknesses in Software and Hardware

Both software and hardware components of modern networks can harbor vulnerabilities. Software vulnerabilities may include coding errors, design flaws, or unpatched software, while hardware vulnerabilities could result from weak authentication mechanisms, firmware weaknesses, or hardware misconfigurations.

Modern networks face vulnerabilities in software and hardware, human factors, and complex architectures. Software and hardware issues include coding errors and weak authentication, while human factors involve errors and social engineering. Complex architectures, like hybrid clouds, add security challenges. Mitigating these requires regular updates, security awareness training, network segmentation, and integrating security into design. Holistic security approaches are essential for network resilience in a dynamic digital environment.

Regular software updates and patch management are critical in mitigating vulnerabilities. Software vendors release patches to address known security flaws, and neglecting to apply these updates promptly can leave networks susceptible to exploitation by malicious actors.

Outdated or unpatched software and hardware present significant risks to network security. Attackers actively target known vulnerabilities, using them as entry points to compromise systems, steal data, or launch attacks. Failure to keep systems up to date can lead to data breaches, service interruptions, and reputational damage.

Human Factors

Human error and social engineering tactics play a pivotal role in network security vulnerabilities. Employees, whether through negligence or manipulation by external actors, can inadvertently compromise network security. Social engineering tactics, such as phishing, rely on human psychology to deceive individuals into divulging sensitive information.

Insider threats, where individuals within an organization misuse their access privileges for malicious purposes, pose a significant risk[160]. Additionally, unintentional security breaches can occur when well-intentioned employees inadvertently mishandle data or fail to follow security protocols.

To mitigate human-related vulnerabilities, organizations must prioritize security awareness training and user education programs. Educating employees about cybersecurity best practices, recognizing social engineering tactics, and fostering a security-conscious culture can significantly enhance network security.

System Complexities

Modern network architectures, including hybrid cloud environments, virtualization, and distributed systems, introduce layers of complexity. While these technologies offer benefits like scalability and flexibility, they also bring new vulnerabilities.

Complex systems can create vulnerabilities, as they are often harder to monitor and secure comprehensively. Misconfigurations, gaps in visibility, and interdependencies among components can be exploited by attackers.

To address the challenges posed by complex network architectures, organizations should prioritize network segmentation, access control, and robust monitoring. Implementing security-by-design principles[9], where security is integrated into the architecture from the outset, can also help simplify and secure network infrastructure.

9: "Secure-by-Design" means that technology products are built in a way that reasonably protects against malicious cyber actors successfully gaining access to devices, data, and connected infrastructure [55].

Vulnerabilities in modern networks are multifaceted and demand a holistic approach to network security. This includes staying current with software and hardware updates, addressing human-related vulnerabilities through education and training, and simplifying network architectures where possible to reduce complexity. By addressing these vulnerabilities, organizations can enhance the resilience of their networks in an ever-evolving digital landscape.

4.3 The Need for Innovative Approaches

Network security is a dynamic field that constantly evolves in response to ever-changing threats and technological advancements. In this section, we will explore the imperative need for

innovative approaches in network security and consider the implications and challenges that accompany this necessity.

In today's digital landscape, traditional network security measures alone are no longer sufficient to protect against the evolving and sophisticated threat landscape. Innovative approaches are essential to stay one step ahead of cyber adversaries. These innovative strategies encompass not only the deployment of cutting-edge technologies but also the development of creative, adaptive, and forward-thinking solutions.

Cybercriminals continuously adapt their tactics, making it imperative for network security to be equally adaptable. Innovative approaches enable security professionals to respond swiftly to emerging threats, adjusting their defenses to match the evolving tactics employed by attackers.

In the dynamic field of network security, traditional measures are insufficient against sophisticated threats, necessitating innovative approaches. These include deploying advanced technologies and developing adaptive solutions. Challenges include economic and social repercussions of breaches, legal and ethical considerations, and evolving technologies like IoT and AI-driven attacks. The future of network security depends on continuous innovation and vigilance in protecting data and digital ecosystems.

Implications and Challenges

Network security breaches have economic and social implications. Data theft, financial loss, and damage to reputation can have severe consequences. Financial losses resulting from breaches can be devastating, and the loss of customer trust can be equally damaging. Moreover, in the age of digital interconnectedness, the social implications of breaches extend to the privacy and security of individuals, whose personal information may be compromised.

Network security is entangled with various legal and ethical considerations. Privacy concerns, driven by data breaches and intrusive surveillance, have led to the enactment of data protection laws like the General Data Protection Regulation (GDPR) in Europe. Ethical considerations encompass the responsible use of data, transparency in data handling, and the ethical responsibilities of organizations to safeguard user information.

Looking ahead, network security faces a myriad of challenges. The future promises a continued evolution of technology, including the expansion of the Internet of Things (IoT), quantum computing, and AI-driven attacks. These technological advancements introduce novel attack vectors and vulnerabilities. Additionally, the increasing sophistication of cyber

threats poses a formidable challenge, demanding innovative strategies to detect, prevent, and respond to attacks effectively.

The need for innovative approaches in network security is paramount in an ever-evolving digital landscape. The economic, social, legal, and ethical implications of network security breaches underscore the urgency of adopting creative and adaptive solutions. To overcome future challenges, network security professionals must continuously innovate and embrace emerging technologies while remaining vigilant in their efforts to protect sensitive data and maintain the integrity of digital ecosystems.

4.4 Advances in Network Security

One of the techniques to protect and advance network security in today's ever-evolving digital landscape is the comprehensive analysis of all network traffic. This proactive approach to cybersecurity has gained paramount importance as cyber threats continue to grow in sophistication and scale. Analyzing network traffic not only allows organizations to detect and mitigate existing threats but also plays a pivotal role in staying one step ahead of emerging risks.

It is due to these dangers and risks that the use of NetFlow technology has become instrumental in network traffic analysis, providing valuable insights into data flows, helping organizations enhance their security postures, and contributing to the ongoing evolution of network security practices.

The NetFlow technology [117] was developed in an attempt to collect information about IP traffic in a simple way, thus being able to track the flows. NetFlow, which allows us to obtain several characteristics of the packets that pass through network devices, has become an industry standard protocol. Nowadays, it is used by the majority of routers in the world[10]. This protocol has several versions but the most widely used standard versions are versions 5 and 9, which gather data such as the source IP, the destination IP or the source and destination ports, among others. One of the consequences of the design of this protocol is that we are unable to inspect the payload of the packets, called deep packet inspection (DPI). Additionally, in order to avoid the saturation of the

10: NetFlow is extensively used in network traffic analysis, with Cisco and other major vendors incorporating it as a default feature in their devices, highlighting its widespread adoption and significance.

routers the Sampled NetFlow is used. This allows the system administrators to define the threshold in which the flows of Netflow are gathered.

As of today, due to the impossibility of exhaustively analyzing each and every one of the network packets, the interest in gathering information based on the network flows is growing dramatically. One of the first topics to be studied was the traffic classification [125]. The accuracy obtained without sampling was 96.67% [25] and the accuracy with sampling was 51.02% [40]. Nowadays, the researchers go further trying to recognize worm attacks [1] or building generic detection systems [292].

One of the latest topics in this research area is the detection of network attacks, such as DDoS or Ports scanning. For this purpose, a large number of papers using machine learning, e.g. KNN, SVM, have been published [275, 42, 257, 256, 129, 234, 37]. However, there are few published research [163] working with the increasingly popular Convolutional Neural Networks, in particular, with the well known architecture called ResNet. That paper describes a methodology with better results than some traditional methods described before, reaching a 95.86% accuracy. Despite that, the paper was written ambiguously, not describing the version of NetFlow used nor the columns selected on the research, among other examples. This method of Convolutional Neural Networks applied in Netflow for detection networks attacks is a new field opened by this group of researchers. Our work continues the research done in this new field working on Sampled Neflow.

4.5 Methodological Exploration

In this section, the core of the research is explained. The different parts of the project are explained separately below, from the generation of images, to the training of the network. We follow a three-step methodology (Figure Figure 4.3). First, data gathering of NetFlow flows is done by using DOROTHEA, a NetFlow dataset generator explained below. Next, in the image generation phase NetFlow data (1D) is converted to images (2D). Finally, the model of CNN is trained, and later validated.

Table 4.1: Datasets of images used to train the models.

	Training		Validation	
Sampling	Attack	Normal	Attack	Normal
-	241.910	236.855	303.907	303.907
500	2742	5506	2727	2727
1000	1428	1428	1292	1292

4.5.1 Data Gathering

In this section, we explain the datasets used in our research. In our work, we have a different datasets for each sampling. The datasets were obtained using DOROTHEA, an open-source tool to gather netflow datasets [37]. We used three types of dataset, the first one without sampling, the next one with a sampling rate of 500 and the last one with 1000. All of them are NetFlow version 5. Datasets were taken in similar duration periods, but as a consequence of sampling, the size of datasets are smaller. The first dataset is used in order to probe that we have a similar accuracy to previous research[163], verifying that the solution of CNN on NetFlow works. We have used a dataset with a sampling rate of 1000 as it resembles a real production environment due to the fact that it is usually the most common sampling rate on big networks. Finally, we chose a sampling rate of 500 because it is a middle value between the real environment and the NetFlow without sampling. In the Table 4.1, the total number of images generated for the different datasets are shown. The dataset without sampling is the bigger one with a total of 1M images separated in training and validation phases. The dataset with a sampling rate of 500 has about 10k images. Finally, the dataset with a sampling rate of 1000, that imitates a real environment, has about 5000 images. All the datasets have a total of 12GB of disk usage. The dataset is publicly available with the project.

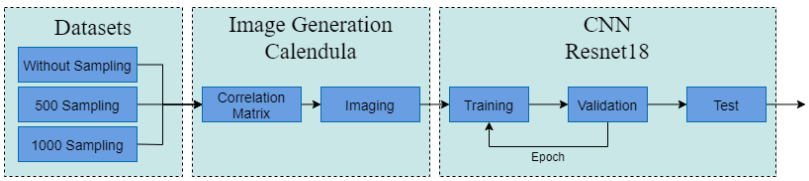


Figure 4.3: Method architecture steps overview.

4.5.2 Image Representation

Imaging is one of the most important aspects of this research field. For the training and testing of the network, it is very important to be able to correctly represent all the flow data, in such a way that the network can be able to catalog the flows. For this task, we have used the parallel computing cluster of Calendula. Calendula is the supercomputer of SCAYLE (Supercomputing Center of Castilla and León) with a total of 397 TFlops which has allowed us to generate a large number of images in a reduced time for the generation of the model. This phase is one of the most critical ones, because it is a process that requires a big amount of time. On our approach, we decided to run 36 processes simultaneously. The creation of the images from the datasets took a few weeks executing on Calendula.

To carry out the task of representing the values properly, only numerical values have been used, discarding other NetFlow features with different types of values. Regarding the eliminated characteristics, in addition to the non-numerical ones, we have also discarded the ones which are related to the exact time in which the flow was extracted, the next hop of the packet or the IP from which the flow was extracted, since they do not provide any type of relevant information for the image. It should be noted, that some columns, e.g. *src_as*, *dst_as*, *src_mask* or *dst_mask*, may include relevant information on datasets retrieved from real infrastructures. The source IP and the destination IP are converted to decimal values before the image is generated. Finally, NetFlow v5 features used in this research are: *dpkts*, *doctets*, *first*, *last*, *srcaddr*, *dstaddr*, *input*, *output*, *srcport*, *dstport*, *prot*, *tos* and *tcp_flags*. Therefore, the official NetFlow v5 columns discarded due to the lack of information are: *unix_secs*, *unix_nsecs*, *sysuptime*, *exaddr*, *engine_type*, *engine_id*, *nexthop*, *src_mask*, *dst_mask*, *src_as*, *dst_as*.

First of all, a correlation matrix is generated with all the characteristics defined before. In order to create the correlation matrix, the Pandas library [177] is used. After that, each of the columns is surrounded by their eight more correlated values making a matrix of 3x3. Finally, all the 3x3 matrices are joined on a big matrix called Surrounding Correlation matrix (SC matrix). This methodology was presented by a group of researchers [163] from Purdue University. Finally, this SC

matrix values are replaced by each of the flows, generating an image similar to Figure Figure 4.4.

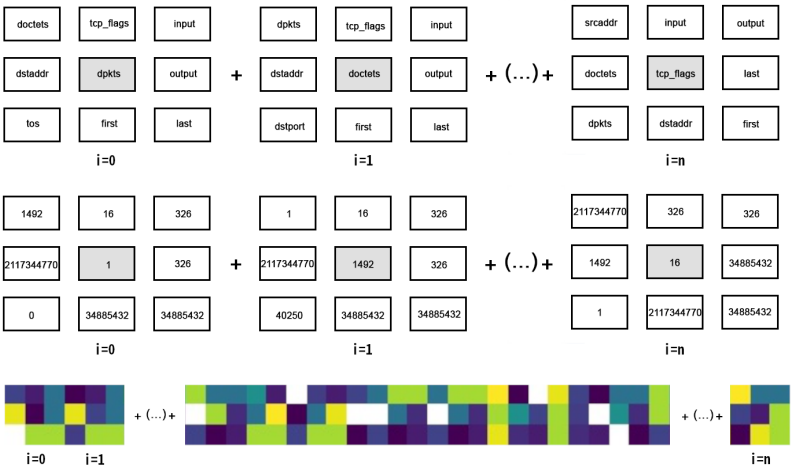


Figure 4.4: Matrix transformation of values obtained from Netflow Traffic.

4.5.3 Training Process

The training phase was done using the popular library PyTorch [203]. This tool is a machine learning library which is focused on usability and speed, allowing us to train ResNet-18. Our approach was to use the model offered by PyTorch, freezing all the layers except the final one. This methodology allowed us to train the model with a maximum performance. The Convolutional Neural Network model used was ResNet-18 [111]. This architecture is a reliable model used in the previous work in this field by Purdue's researchers. ResNet-18 is well-known to be used for classification [111].

Before feeding the model with data, the images need to be resized to 224x224 in order to fit the requirements of the network[11]. After that, the last layer of the model is trained using a defined number of epochs in order to get the best accuracy possible. In each of the epochs there are two phases defined, training and then validation phase. This technique lets us save the best model.

11: Images are resized to 224x224 pixels to meet the specific input dimension requirements of CNN architectures like VGG16, ensuring consistent tensor size for effective feature extraction and neural network processing.

4.6 In-Depth Analysis of Findings

In this section, a detailed overview of the results are presented. The results are obtained using the methodology explained

above. First of all, the accuracy of Netflow without sampling is 96.58%. This accuracy, that is greater than the previous work [163], demonstrates that Convolutional Neural Network (CNN) could be used on Netflow with good results. Moreover, with this dataset, we have double-checked the previous research. This dataset is the bigger one, so the training of the model was the largest in terms of time. In other words, the training with this dataset delays approximately 11 hours. The accuracy of Convolutional Neural Network (CNN) is similar to any other method used in the literature, e.g. KNN, SVM [275, 42, 257, 256, 129, 234, 37]. In the table 4.2 the accuracies of other classifiers are shown.

Classifier	Accuracy
CNN	96.58%
KNN	96.41%
LR	94.83%
SGD	92.10%
OvR	93.23%
CART	90.18%
RF	90.18%
AB	90.18%
BRBM	78.22%
QDA	51.71%
LDA	51.71%
NB	51.71%
BC	51.71%

Table 4.2: Accuracy from the literature.

The second dataset contains the data with a sampling rate of 500. Like we explained before, this sampling rate is a middle value between the actual literature and the real environments. Using this second dataset, the accuracy was 94.15%, decreasing a 2.52% below the results without sampling. This value shows that Convolutional Neural Network (CNN) could be applied successfully on Sampled NetFlow.

The last one is the simulation of a real environment with a sampling rate of 1000. This is the sampling rate that is normally applied in real routers to avoid the degradation of the performance due to their saturation. The accuracy obtained by the model with this dataset is slightly disappointing, getting a 50.11%. This supposes a 48.12% decrease below the

The study presents results using a detailed methodology, achieving 96.58% accuracy with Netflow without sampling, demonstrating CNN's effectiveness. Training on the largest dataset took 11 hours. CNN's accuracy is comparable to other methods like KNN, SVM. However, accuracy drops to 94.15% with a 500 sampling rate dataset and significantly to 50.11% with a 1000 rate, highlighting challenges in applying CNN to sampled NetFlow in real environments and suggesting the need for further research. The decrease in accuracy is attributed to data loss in higher sampling rates.

first dataset and a 46.78% below the second one with 500 of sampling rate. Although the results are not encouraging, they open an opportunity to research new methodologies in this way.

In summary, the accuracy decreased drastically when the rate of Sampled NetFlow is greater than 500. The reason for that is the amount of information that is lost when the NetFlow is sampled. In the Figure Figure 4.5 the accuracies of the datasets are shown.

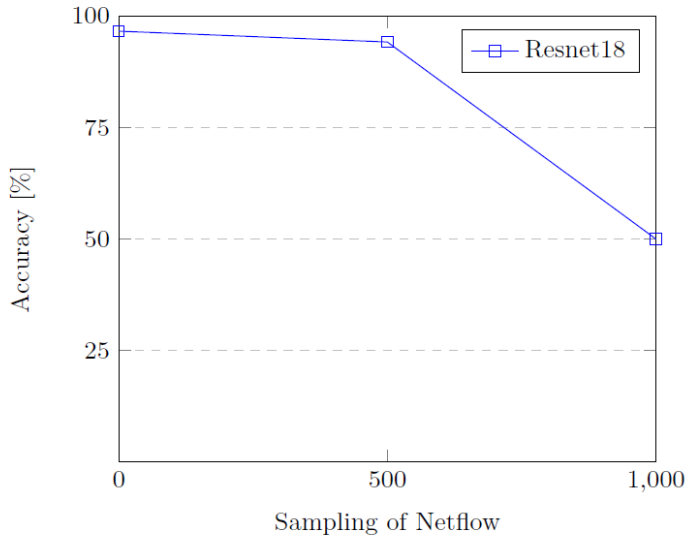


Figure 4.5: Resnet18's accuracy steeply declines from near-perfect to below 50% as Netflow sampling increases from 0 to 1,000.

4.7 Results and Future Directions

This experiment presents the first work using Convolutional Neural Networks on Sampled Netflow. Moreover, in this research we describe the limitations that Convolutional Neural Networks have on Sampled NetFlow. This is an issue that also appear in other similar approaches [40]. Using Convolutional Neural Networks we obtain a 94.15% of accuracy with a sampling rate of 500. We believe these insights can be the foundation for more systematic future works in this field.

This work shows that Convolutional Neural Networks (CNN) accuracy decreases when the interval of Sampled Netflow increases. Furthermore, with a simulated real environment dataset with a sampling rate of 1000, the accuracy is greatly decreased. However, there are smaller networks where the sampling rate is 500 or less in which this work could be applied. On a future work, the model is going to be tested

with real data to compare the accuracy obtained. Besides, another line of research is the study of the information added to the model for each of the features of NetFlow. In this line of work, different architectures of CNN or different methods of training could be studied. Furthermore, the performance of the image creation process is critical in order to implement this methodology on a real environment, so this could be a future limitation of this new approach.

Acknowledgement

The research described in this article has been partially funded by Instituto Nacional de Ciberseguridad de España (INCIBE), under the grant "ADENDA 4: Detección de nuevas amenazas y patrones desconocidos (red Regional de Ciencia y Tecnología)", addendum to the framework agreement INCIBE–Universidad de León, 2019–2021; and by the Spanish Ministry of Science, Innovation, and Universities RTI (RTI2018-100683-B-I00) grant.

PE Malware Analysis: DNN Exploration

5

5.1 Fundamentals of Portable Executable (PE)

The Portable Executable (PE) format is a file format for executables, object code, and DLLs (Dynamic Link Libraries[1]) in 32-bit and 64-bit versions of Windows operating systems. This format is a cornerstone in the Windows operating environment, playing a crucial role in the loading and execution of software applications.

At its core, a PE file consists of a header that includes critical information for the Windows loader: the machine type, sections of the file, and the entry point of the code. Following this header are sections like .text, .data, and .rdata, which contain the executable code, initialized data, and import/export information, respectively. This structured approach in PE files allows for efficient execution and resource management, making them integral to Windows operating systems.

PE files are adaptable and versatile, capable of housing a wide array of information necessary for the execution of different types of applications. They are the standard format for Windows executable files, making them ubiquitous in any Windows-based computing environment.

1: DLLs are modular files containing code and data used by multiple programs simultaneously.

5.1.1 Current State of PE Files in the Context of Software and Malware

While PE files are fundamental to legitimate software operations, their ubiquity and complex structure make them a prime target for exploitation by malware authors. Malware in PE format can appear benign to users and systems, allowing it to bypass initial security checks and execute malicious code within a trusted environment.

The current landscape sees a persistent prevalence of PE files being used as a vehicle for malware delivery and execution. This situation is exacerbated by the evolving sophistication

PE files, crucial for legitimate software, are frequently exploited for malware due to their complexity and ubiquity. Malware authors use techniques like obfuscation and encryption in PE files, challenging detection efforts. This widespread issue underscores the need for sophisticated detection methods, like Deep Neural Networks, in Windows environments.

of malware creators who continually devise new methods to exploit the intricacies of the PE format. Techniques such as code obfuscation, packing, and encryption are commonly employed to hide malicious intent, making detection increasingly challenging[158].

The use of PE files in malware distribution is not limited to obscure or rarely used software; it spans a wide range of commonly used applications and tools. This widespread use poses significant risks to users and systems, highlighting the need for advanced detection and analysis techniques. Additionally, there are various types of PE malware, each with different objectives. This includes infostealers, which aim to extract sensitive data, downloaders that fetch additional malicious software, and backdoors, which provide remote unauthorized access to infected systems. These diverse malware types further complicate the threat landscape, necessitating nuanced and sophisticated security approaches (Figure 5.1).

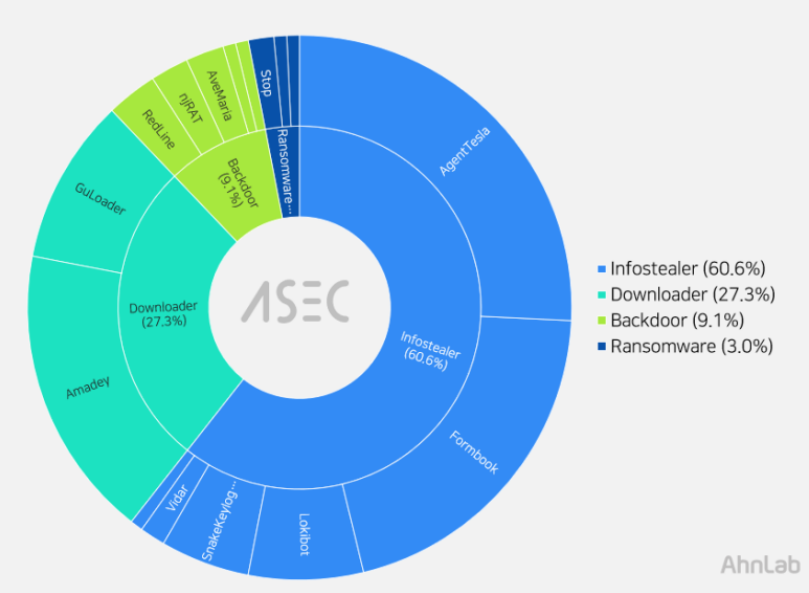


Figure 5.1: Example of the distribution of different types of malware in PE files by ASEC automatic analysis system[21], categorized malware from May 1st to 7th, 2023. The statistics show infostealers as the most prevalent at 60.6%, followed by downloaders (27.3%), backdoors (9.1%), and ransomware (3.0%).

The PE format's role in both legitimate computing and malware presents a dual-faceted challenge. Understanding its structure and operation is key to developing effective strategies for malware detection and prevention, particularly in Windows environments. This sets the stage for exploring advanced methodologies, like Deep Neural Networks (DNNs), in the ongoing battle against PE-based malware.

5.2 Historical Evolution of Malware in PE Files

5.2.1 Early Instances of Malware in PE Files

The history of malware targeting Portable Executable (PE) files is as old as the format itself, dating back to the early days of Windows operating systems. Initially, malware in PE files was relatively straightforward, often involving basic viruses that replicated themselves by attaching to legitimate PE files. These early forms of malware were primarily focused on proliferation rather than sophisticated harm or data theft.

One of the earliest notable examples was the Win32 virus[20], which emerged in the mid-1990s. This virus marked a significant shift in malware development, exploiting the vulnerabilities of the Windows 95 and Windows NT operating systems. It utilized the PE format to execute malicious payloads while camouflaging itself within legitimate software[264], setting a precedent for future malware.

These early instances of PE malware were relatively easier to detect and mitigate, as they often contained recognizable patterns and did not employ advanced evasion techniques. However, they set the stage for more complex malware, evolving alongside advancements in Windows operating systems and security technologies.

5.2.2 Modern Trends and Developments in PE Malware

In recent years, the landscape of PE malware has undergone significant evolution, becoming more sophisticated and harder to detect. Modern PE malware employs a range of advanced techniques to evade detection and enhance its malicious impact.

Recent years have seen PE malware evolve with advanced evasion techniques like packing, obfuscation, and polymorphism, making detection harder. Modern PE malware includes sophisticated functionalities like rootkits and exploit kits, targeting Windows systems and creating significant cybersecurity challenges. This evolution necessitates advanced detection methods like Deep Neural Networks.

One of the key trends in contemporary PE malware is the use of packing and obfuscation. Malware authors often pack PE files with layers of encryption and obfuscation to conceal malicious code from antivirus software. Such techniques make static analysis of PE files increasingly challenging, requiring more dynamic and intelligent analysis methods.

Another prevalent trend is the use of polymorphic and metamorphic malware[231, 153]. These types of malware can change their code or structure with each infection, making signature-based detection nearly impossible. This adaptability enables malware to persist within systems and networks, evading traditional antivirus solutions.

Additionally, the integration of advanced functionalities in malware, such as rootkits, backdoors, and trojan capabilities, has made PE malware a potent threat. These functionalities allow attackers to gain deep control over infected systems, steal sensitive information, and create botnets for coordinated attacks.

The use of exploit kits[2] targeting specific vulnerabilities in Windows operating systems is also a notable development. These kits are often sold in underground markets, making sophisticated attack capabilities accessible to a broader range of malicious actors.

2: Malware exploit kits are prepackaged software tools that automate the exploitation of vulnerabilities in software to facilitate malware distribution and attacks.

The evolution of malware in PE files reflects a continuous arms race between attackers and defenders. From simple viruses to complex, multi-functional threats, PE malware has become a significant concern in cybersecurity. Understanding its historical progression and current trends is crucial in developing effective countermeasures, paving the way for the exploration of advanced techniques like Deep Neural Networks (DNNs) in malware detection and analysis.

5.3 Overview of Deep Neural Networks (DNNs) in Malware Detection

Deep Neural Networks (DNNs) represent a significant advancement in the field of artificial intelligence, especially in the realm of machine learning.

In the context of malware detection, DNNs offer a powerful tool for identifying sophisticated threats. Traditional malware detection methods often rely on signature-based techniques that struggle to keep pace with rapidly evolving malware. In contrast, DNNs can learn and identify patterns indicative of malware, even in the absence of known signatures, making them adept at detecting novel or previously unseen malware variants.

5.3.1 DNN Methods in Malware Analysis

Several types of DNN architectures are particularly relevant to malware analysis, each method's strengths and weaknesses become evident:

In malware analysis, various DNN architectures like RNNs, Autoencoders, CNNs, and GANs each offer unique strengths. RNNs analyze sequential data like opcode patterns, Autoencoders detect anomalies in 'normal' PE files, CNNs process binary data as images identifying spatial patterns, and GANs generate synthetic malware samples for robust model training. Each method contributes differently to combating malware threats in digital environments.

- Recurrent Neural Networks (RNNs): RNNs excel at processing sequential data, making them ideal for analyzing time-series data or textual content. In the context of PE malware, RNNs can be used to analyze the sequential flow of opcode or API calls, learning patterns that are indicative of malicious behavior.
  - Strengths: RNNs excel in analyzing sequential data, such as opcode patterns within PE files, capturing temporal dependencies that other networks might miss.
  - Weaknesses: RNNs can suffer from issues like vanishing or exploding gradients, especially in dealing with long sequences, potentially impacting training efficiency.
- Autoencoders: Autoencoders are a type of unsupervised learning technique used to learn efficient codings of input data. In malware detection, autoencoders can be employed to learn a representation of "normal" PE files and then identify deviations from this norm, which could signal the presence of malware.
  - Strengths: Autoencoders are adept at dimensionality reduction and anomaly detection, useful for identifying unusual patterns in PE files that could indicate malware.
  - Weaknesses: They may not be as effective in a supervised learning context and could require substantial training data to define "normal" effectively.
- Convolutional Neural Networks (CNNs): CNNs are particularly well-suited for processing data with a grid-like topology, such as images. In malware analysis, CNNs can be used to analyze the binary content of PE files, treating the binary data as an image. This approach allows CNNs to capture the spatial relationships between different parts of the file, identifying patterns that might indicate malicious content.

  - Strengths: CNNs are highly effective in identifying spatial hierarchies and patterns in data, making them suitable for analyzing the binary structure of PE files.
  - Weaknesses: CNNs may require significant computational resources, particularly for training, and may struggle with very large input sizes or highly variable file structures.

- Generative Adversarial Networks (GANs): Though primarily used in generative tasks, GANs can also be applied in malware detection. They can generate synthetic malware samples for training, improving the robustness of the DNN models against novel or evolving malware strains.
  - Strengths: GANs can generate new malware samples for training, enhancing the model's ability to identify novel malware types.
  - Weaknesses: The complexity of GANs can lead to challenging training processes and the requirement for large and diverse datasets.

Each of these DNN methods offers a unique approach to understanding and detecting malware in PE files. By leveraging the strengths of these varied architectures, researchers and cybersecurity professionals can develop more effective tools to combat the ever-evolving threat of malware in today's digital landscape.

5.3.2 Assessment Criteria for DNN Methods in Malware Detection

In evaluating the efficacy of various Deep Neural Network (DNN) methods for detecting Portable Executable (PE) malware, it's crucial to establish a set of clear, measurable criteria. These criteria allow for an objective comparison of different DNN approaches. The primary criteria include:

- Accuracy: The ability of the DNN method to correctly identify malware without mistakenly flagging benign software. This includes measuring the true positive rate (sensitivity) and true negative rate (specificity).
- Efficiency: This encompasses the computational resources and time required for the DNN method to

analyze and classify PE files. Efficiency is particularly important in environments where real-time or near-real-time malware detection is necessary.
- Robustness: The resilience of the DNN method against various obfuscation techniques used by malware developers, such as packing, encryption, and polymorphism.
- Scalability: The capability of the DNN method to maintain performance as the volume of data increases, which is crucial for adapting to the growing amount of PE malware.
- Adaptability: The ability of the DNN to learn from new malware samples and adapt to evolving malware tactics and trends.

5.4 Implementing DNNs for PE Malware Detection

In the ongoing battle against malware, particularly those targeting Portable Executable (PE) files, the need for advanced and effective detection methods is paramount. My research experiment focuses on the implementation of Deep Neural Networks (DNNs) for the detection of PE malware, a decision driven by several compelling factors that highlight the potential of DNNs in this domain.

The research experiment focuses on using Deep Neural Networks (DNNs) for detecting PE malware, leveraging their ability to handle complex data, adapt to evolving threats, effectively recognize patterns, and reduce false positives and negatives. This study aims to test DNNs' capabilities against modern malware challenges, contributing to cybersecurity advancements.

- Capability to Handle Complex Data Structures: PE files, with their intricate and layered structure, present a significant challenge for traditional malware detection methods. DNNs, known for their ability to process and learn from complex and high-dimensional data, are well-suited to analyze the nuanced patterns and structures within PE files.
- Adaptability to Evolving Threats: The landscape of malware is constantly evolving, with new variants emerging regularly. DNNs have a proven track record in learning from new data and adapting to changing patterns, a critical feature for staying ahead in the malware detection arms race.
- Effectiveness in Pattern Recognition: Malware within PE files often exhibits subtle and sophisticated patterns that evade traditional signature-based detection methods. DNNs excel in identifying and learning these

complex patterns, making them highly effective for identifying both known and unknown malware types.
- Reduction in False Positives and Negatives: One of the challenges in malware detection is maintaining a balance between sensitivity (true positive rate) and specificity (true negative rate). DNNs, through their nuanced understanding of data, have the potential to reduce false positives and negatives, thereby improving the reliability of malware detection.

This research experiment is designed to rigorously test the capabilities of DNNs in detecting PE malware, aiming to contribute valuable insights and advancements to the field of cybersecurity. By addressing these objectives, the experiment seeks to validate the hypothesis that DNNs offer a superior approach to PE malware detection, capable of meeting the challenges posed by modern malware threats.

5.4.1 Methodology and Experiment Design

In this experiment, Deep Neural Networks (DNNs) have been used, which are a class of artificial neural networks that contain multiple layers of nodes between the input and output layers. These layers are typically fully connected, meaning that each node in one layer is connected to every node in the next layer.

DNNs are trained using a process called backpropagation[100], which involves computing the gradient of a loss function with respect to the network's parameters and adjusting the parameters in the direction of the negative gradient. This process is typically performed using stochastic gradient descent[3], which iteratively updates the parameters using small batches of training data.

3: Stochastic Gradient Descent is an iterative optimization algorithm that updates parameters randomly selected from data points to minimize a loss function.

DNNs have been shown to be highly effective in a wide range of applications[119], including image and speech recognition, natural language processing, and robotics. By incorporating multiple layers of nodes, DNNs are able to learn increasingly complex features of the input data, allowing them to model more complex functions than traditional neural networks. However, training DNNs can be challenging due to issues such as overfitting[27], vanishing gradients, and exploding gradients, and often requires specialized techniques such as

regularization, weight initialization, and batch normalization.

A Deep Neural Network (DNN) is a multi-layer perceptron with many hidden layers, referring to the layers of nodes between the input and output layers, that may be started using the DBN pre-training technique [228]. DBN pre-training is a technique where each layer of a deep neural network is trained separately in an unsupervised manner using a restricted Boltzmann machine, before fine-tuning the entire network with backpropagation, resulting in better performance on downstream tasks. If the number of layers and units in a single layer is raised, they can express functions of greater complexity. Deep Learning techniques can assist people in establishing mapping functions for operation convenience if they have adequate labeled training datasets and acceptable models [162]. The reason for selecting deep neural networks as a method for malware detection is twofold: firstly, the abundance of available samples, and secondly, the potential for achieving high accuracy.

In our experiment, we have designed a sequential model consisting of seven layers. All of them are dense layers, simple layers of neurons where each neuron receives input from all the neurons in the previous layer, hence the term dense. We have defined the hidden layer's size as 50 nodes/neurons for each layer[4].

4: A hidden layer in a neural network processes inputs from previous layers, extracting features without direct exposure to external inputs or outputs.

The Rectified Linear Unit (ReLU) has been used as activation function. It is a commonly used function that returns the input if it is positive, and produces zero otherwise. In mathematical terms, the ReLU function can be defined as shown in Equation 5.1, where x is the input to the neuron, and max(0, x) returns the maximum value between 0 and x.

$$f(x) = max(0, x) \tag{5.1}$$

The ReLU activation function is popular because it is computationally efficient and can help to avoid the vanishing gradient problem that can occur with other activation functions like "sigmoid" or "tanh". The vanishing gradient problem occurs when the gradient of the activation function becomes very small, leading to slow training or convergence issues. By using the ReLU function, the output of the neuron becomes either the input itself (if it is positive) or zero (if it is negative).

This allows the network to learn a sparse representation of the data, which can be helpful for reducing overfitting and improving generalization. Using the ReLU activation function can help improve the performance of neural networks by providing a simple, efficient, and effective way to introduce nonlinearity into the network.

For the output layer, we used the Softmax activation function that converts the outputs of the previous layer, which can be any real number, into a probability distribution over the predicted classes, in our case Benign or Malware. The Softmax function takes a vector of real numbers as input and normalizes them so that the sum of the values in the output vector is equal to one. Each element in the output vector represents the probability of the input belonging to a particular class. The softmax function is defined as shown in Equation 5.2, where x_i is the ith element of the input vector x, and the sum is over all elements in x. The function returns a vector of the same dimensionality as x, where each element is in the range (0,1) and the sum of the elements is 1.

$$Softmax(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \tag{5.2}$$

The model is compiled using the Adam optimizer, which is a popular algorithm used for optimizing the weights of neural network models during training. It is a type of stochastic gradient descent algorithm that updates the weights based on the gradient of the loss function with respect to the weights. The Adam optimizer uses a combination of adaptive learning rate methods and momentum to optimize the weights efficiently. It adapts the learning rate of each weight individually based on the history of its gradient updates. This helps to improve the convergence speed and stability of the training process, especially for large datasets or complex models.

In addition to setting the optimizer, other parameters are specified during model compilation, such as the loss function to optimize and the metrics to evaluate the performance of the model during training. We used the sparse categorical cross-entropy, which is a commonly used loss function in neural networks. It is a variant of the categorical cross-entropy loss function, which is used when the target variable is in one-hot encoded format.

The formula for the sparse categorical cross-entropy loss function is similar to that of the categorical cross-entropy, except that it takes the integer-encoded labels as input instead of the one-hot encoded labels. The loss is defined as shown in Equation 5.3, where y is the true label (in integer format), $\hat{y}$ is the predicted probability distribution, and the sum is over all classes.

$$-\Sigma[y * log(\hat{y})] \tag{5.3}$$

As the last step of this experiment, in order to train the model, a batch size of 100 and a maximum of 20 epochs have been defined.

Data Gathering

Malicious software was detected using a dataset of over 100,000 executable files, which were labeled as benign and malware. The dataset was obtained from VirusShare[262] which is a repository of malware samples to provide security researchers, incident responders, forensic analysts, and the morbidly curious access to samples of live malicious code. This repository has more than 55,423,284 malware samples uploaded by the community. To obtain a comprehensive set of pertinent features of malware, including Headers, Extracted Functions, Sections, and others, and to accurately classify all samples, we used VirusTotal. Using its API and uploading each sample, the file is scanned. This scan returns a total of 12 flags with relevant information about each sample. As a result of this scan, files with more than 10 detection flags are classified as malware, while all other files are recognized as safe. This threshold is defined because the platforms themselves define a file as potentially malicious once this number is exceeded.

The flags contained in each of the files after scanning are as follows:

1. Load Configuration: Identifies whether the file includes specific configuration settings for system loading processes.
2. Debug Section: Detects presence of a section in the file containing debugging information created during compilation.

3. Has Exceptions: Checks if the file utilizes exception handling mechanisms in its operation.
4. Has Exports: Determines whether the file exports certain symbols or functionalities to other programs or processes.
5. Has Imports: Identifies if the file imports symbols or functionalities from external sources or libraries.
6. NX Bit: Examines use of the NX bit for segmenting memory and marking regions as non-executable in the CPU.
7. Has Relocations: Verifies the presence of relocation entries that guide updates to section data when needed.
8. Resource Usage: Checks if the file consumes system resources, such as memory or processing power.
9. Rich Header: Detects a rich header section indicating the compilation environment of the Windows executable.
10. Digital Signature: Determines whether the file is digitally signed, indicating authenticity and integrity.
11. TLS Usage: Checks for the usage of Thread Local Storage (TLS) in the file's execution.
12. Entry Point Bytes: Contains the standardized initial bytes of the file's entry point function for execution start.

Once the samples are marked as benign or malware, in our experiment we will use the headers of each executable PE to build our model. The headers are data structures located at the beginning of the file that contain crucial information about how the operating system and the program should interpret and execute the file. Headers provide details about the system architecture, file type, memory sections, start addresses, used shared libraries, among other important aspects.

The dataset from VirusTotal, containing PE file information, undergoes cleaning where file hashes are removed to prevent issues during shuffling. Data is normalized, scaling input features to zero mean and unit variance for effective neural network learning. The dataset includes recent samples, with 50% classified as malware in both training (80%) and testing (20%) sets.

Data Pre-Processing

The information dump from Virustotal has certain characteristics regarding the information of the Portable Executable (PE) files. To generate our dataset, a cleaning process is carried out on this data. As part of this process, one of the characteristics contained in the data, which is the HASH of the PE files, is removed. This is because if the hash of each file in the dataset is used as a unique identifier, shuffling the

data would cause the order of the file hashes to change as well. As a result, any downstream process that relies on the file hashes, such as data verification, would be affected by the shuffling.

Normalization is a common preprocessing technique used in deep learning to transform input data into a common scale so that the neural network can learn more effectively. The technique used scales the input features such that they have zero mean and unit variance. This is done by subtracting the mean of each feature from the data points and then dividing by the standard deviation of each feature. The resulting scaled data has a mean of zero and a standard deviation of one.

To create our dataset, we filtered by recently added samples, in order to have an up-to-date dataset with the latest malware. More specifically, 50% of the occurrences in both the train (80%) and test (20%) sets are classified as malware.

5.4.2 Results, Analysis, and Discussion

The results obtained from the experiment of Malicious Software Detection demonstrate favorable outcomes, as evidenced by the accuracy, area under the receiver operating characteristic curve (AUC), and F1 score. These results are shown in the Table 5.1). It should be noted that the dataset used contains only data classified as malware or benign, without taking into account the existence of different subcategories of malware, which implies a greater difficulty of classification. The accuracy, which measures the proportion of correctly classified samples, indicates a high level of precision in the model's predictions. Additionally, the AUC, which quantifies the model's ability to differentiate between positive and negative samples, reveals a strong discriminatory power in identifying malicious software. Finally, the F1 score, which balances the model's precision and recall, suggests a robust performance in both correctly identifying malicious samples and minimizing false positives. Taken together, these metrics indicate that the deep learning approach employed in this study holds significant potential for the detection of malicious software, with implications for enhancing the security of computing systems and networks.

Furthermore, to assess the performance of the deep learning model in predicting the presence of malicious software, a confusion matrix was generated, which is presented in Figure 5.2. This matrix displays the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) generated by the model, based on a comparison between the predicted and true labels of the test dataset.

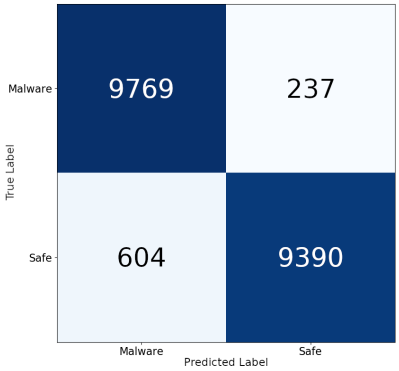


Figure 5.2: PE files Malware detection confusion matrix

In addition to the evaluation metrics mentioned above, a correlation matrix was generated to further investigate the relationship between the features utilized in the deep learning model. The features used to create our model are the headers of each executable file as described in previous sections. The correlation matrix, which is presented in Figure Figure 5.3, displays the pairwise correlations between each feature, with values ranging from -1 to 1. This matrix can aid in identifying any highly correlated features, which can cause issues such as overfitting or multicollinearity. Additionally, the matrix can provide insights into the relative importance of each feature in predicting the presence of malicious software. The interpretation of the correlation matrix is discussed in further detail in the subsequent sections, along with its implications for the performance of the deep learning model.

Table 5.1: PE Files Malware Analysis Results

Experiment	F1	AUC	F1 Accuracy
PE Files detection	0.93	0.98	0.91

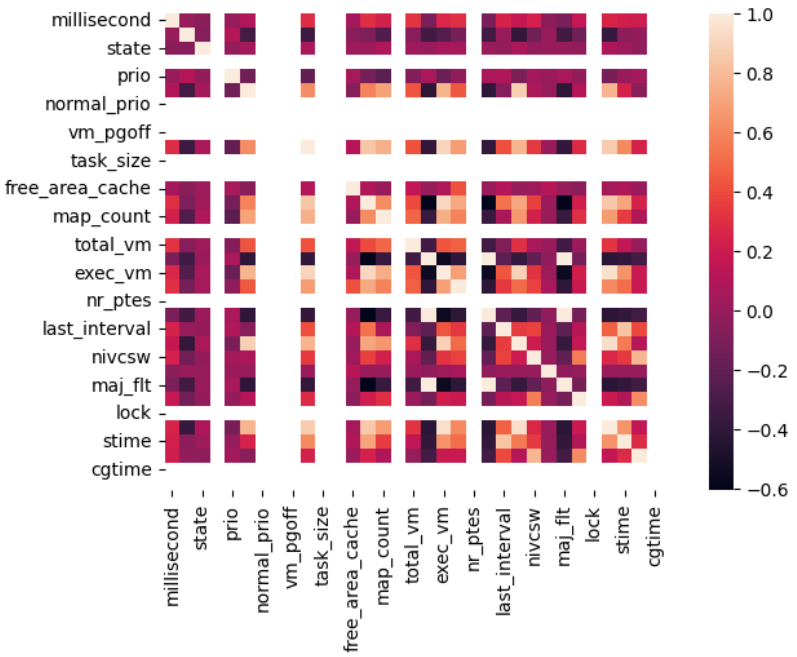


Figure 5.3: PE Headers Features Confusion Matrix

The experiment carried out for malware detection showed favorable results due to an accuracy of 0.91. It should be noted that a positive aspect of this experiment was the large number of samples used for training, but a negative aspect was the fact that all the malware subcategories were unified into a general one, which is a difficulty due to the differences between samples. As shown in related work, there are a wide variety of different approaches, but in our case, we wanted to test by building a simple architecture which, despite this, has achieved positive results. This shows the compatibility of applying deep learning techniques in this scenario, which greatly reduces the effort in categorizing software as malware.

5.5 Integrating DNN Insights into Future PE Malware Defense Strategies

The exploration and implementation of Deep Neural Networks (DNNs) in the detection of Portable Executable (PE) malware, as presented in this thesis, have yielded significant insights and promising results. This concluding section synthesizes the key findings from both the theoretical analysis

of DNNs and the practical outcomes of the experimental research conducted.

Experimenting with using DNNs for detecting PE malware reveals their advanced pattern recognition capabilities, useful for identifying complex malware variants. Experiments show that DNNs effectively differentiate between benign and malicious PE files with high accuracy, and also demonstrate their efficiency and adaptability.

A crucial insight from the DNN analysis is the affirmation of their advanced pattern recognition capabilities. DNNs demonstrated a profound ability to discern intricate and obscured characteristics in PE files, which are often leveraged by malware. This capability is especially pertinent in detecting sophisticated malware variants that traditional, signature-based methods fail to identify.

The experimentation with DNNs further reinforced their suitability for PE malware detection. The models effectively distinguished between benign and malicious PE files, showcasing high accuracy levels. This success is attributable to the depth and complexity of the neural networks, which allowed for a nuanced understanding of the data, far surpassing the capabilities of more rudimentary analytical tools.

Efficiency, another critical aspect, was addressed in the experiments. The DNN models processed data at a pace conducive to real-time applications, suggesting their viability in operational cybersecurity environments. Moreover, the adaptability of DNNs was evident in their ability to learn from new and evolving malware samples, underscoring their potential in addressing the dynamic nature of cyber threats.

5.5.1 Future Implications for PE Malware Detection and Prevention

The findings on Deep Neural Networks (DNNs) in PE malware detection have significant future implications. DNNs' scalability and adaptability make them suitable for diverse, large-scale environments and sustainable against evolving malware. This success encourages broader AI application in cybersecurity, suggesting potential integration with automated threat intelligence and real-time analysis for comprehensive strategies.

The findings from this thesis have profound implications for future strategies in PE malware detection and prevention. Firstly, the integration of DNN models into existing cybersecurity frameworks can significantly enhance the detection of complex malware. This integration involves not just the adoption of the technology but also a shift towards more data-driven, adaptive security methodologies.

Secondly, the scalability of DNNs opens up possibilities for their application in diverse and large-scale digital environments. As the volume of data and the sophistication of malware continue to grow, scalable solutions like DNNs become increasingly crucial.

The adaptability of DNNs also suggests a long-term viability in the cybersecurity domain. As malware evolves, so too

can the DNN models, through continuous learning and adjustment to new threats. This attribute positions DNNs as a sustainable solution, capable of adapting to the ever-changing landscape of cyber threats.

The successful application of DNNs in PE malware detection invites broader exploration and application of advanced AI techniques in cybersecurity. The potential for DNNs to be integrated with other emerging technologies, such as automated threat intelligence and real-time data analysis systems, could pave the way for more comprehensive and proactive cybersecurity strategies.

The exploration of DNNs in this thesis not only demonstrates their immediate benefits in combating PE malware but also opens avenues for future research and development. The insights gained lay a foundation for more advanced, efficient, and adaptable cybersecurity solutions, aligning with the evolving nature of digital threats in an increasingly interconnected world.

Content Filtering: Adult Imagery

6

6.1 Surveying the Adult Content Landscape

The field of cybersecurity and artificial intelligence faces significant and increasingly complex challenges in the digital age, especially in the context of adult content. The proliferation and accessibility of such content on the web has raised a number of concerns related to privacy, security and ethics. Associated dangers include non-consensual exposure and misuse of images, as well as the spread of illegal and harmful material. Artificial intelligence, while offering advanced tools for the detection and filtering of inappropriate content, has also facilitated the creation and distribution of such content, often blurring the lines between the real and the virtual. This complex landscape demands a thorough exploration of current technologies, their applications and limitations, in order to develop more effective and responsible strategies for managing adult content in the digital age.

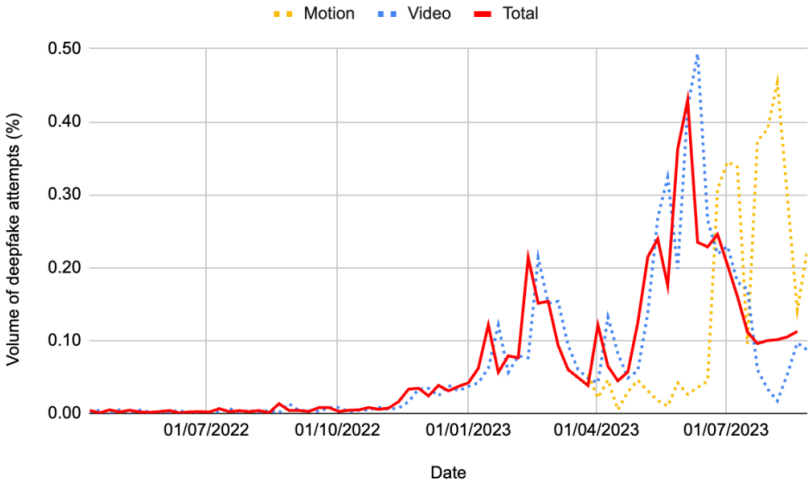


Figure 6.1: Onfido's report focused on the spread of deepfakes and the results are very clear: more than 3000% of deepfakes were detected in the last year compared to 2022. An increase that, according to Onfido, is closely linked to generative AI[198].

In terms of current developments in explicit adult content, significant use of artificial intelligence to generate such images has been observed, also known as DeepFakes [107]. For example, the artificial intelligence "Unstable Diffusion" [121] can generate adult content with anime characters. The Onfido report reveals a 3000% increase in deepfakes over the past year compared to 2022 (Figure 6.1), largely attributed to

generative AI. Factors driving this surge include advanced generative AI like DALL-E, easy access to AI applications for creating deepfakes online at low cost, simplicity in execution, and medium-complexity attacks in large quantities yielding high overall gains[226].

This development has generated controversy, especially in relation to the lack of safety filters, which could allow the processing of realistic images of well-known people or celebrities in a harmful way. In addition, the generation of unrestricted content may be part of a malicious system for generating images in fake news, thus affecting the dissemination and amplification of misinformation.

6.1.1 Deepfakes Hazard

The development of Deepfake technologies has primarily advanced through deep learning, with a focus on face replacement in videos using data-rich images of well-known figures. Initially used for impersonating public figures and politicians, these technologies have evolved significantly. Early methods in 2017 used CNNs for static images, followed by generative networks considering time sequences. FaceswapGAN and autoencoder networks later improved video realism. FaceShifter introduced advanced techniques for realistic movements. In face recreation, techniques control expressions in images, evolving with computing advances and integrating features like RNNs for mouth movements and Spatio-temporal architectures for complete video outputs, enhancing realism and continuity.

The development of Deepfake-focused technologies has extended mainly to advances in deep learning algorithms. Many of these technologies have focused primarily on replacing faces in videos from images, mainly with well-known characters, since a comprehensive set of data is available with which the algorithm achieves a higher performance. Although DeepFake technology can be utilized positively in various applications such as cinema, virtual, and immersive environments, its predominant usage since its inception in 2017 has been focused on impersonating public figures, including politicians and even adult content actors and actresses [71, 84]. Since then, applications such as FakeApp, FaceSwap and Deepnude began to spread. In all cases, the target has not only pursued the violation of the privacy of public figures but they have been used in different political campaigns to influence the public opinion of voters. Depending on the objectives of these algorithms mainly, they can be divided into two broad categories: (i) face exchange and (ii) face recreation.

In the category of face exchange, the methods used appear mainly in 2017 [144], where Convolutional Networks (CNNs) were used. These early approaches worked well mainly with static images, as the motion was not considered if they were more focused on manipulating high-resolution photos. That same year, Olszewski[196] proposed a new approach that did take the sequence of time into account, using a deep generative network, which are networks designed to

generate new data that is similar to the training data it was trained on, only needed a video and an image in RGB format. Subsequently, the first videos of impersonated public figures began to increase, which were the letter of presentation of the potential for negative uses of developed methods.

In the following years, Faceswap-GAN[166] was presented, in which two neural networks worked together in an adversarial manner to generate realistic facial swaps, considerably improving the performance of the generated videos. Later, autoencoder networks[242], networks that compress data using an encoder and reconstruct the original input using a decoder, were added in tools such as VGGFace. An open DeepFaceLab, open-source deep learning framework used for face swapping and face restoration in images and videos, was developed [159], allowing people without specific knowledge of neural networks or artificial vision techniques to generate DeepFakes. There is a disruption in the state of the art.

Unlike the methods presented so far, in which mainly static images were incorporated into videos, FaceShifter [150] introduced attributes extracted from the faces to achieve more realistic movements. Specifically, AEI-Net and HEAR-Net promoted the integration of these movements[1]. The results obtained with these latest techniques have shown a tremendous qualitative leap, generating very difficult to distinguish even for a human being.

1: AEI-Net is an autoencoder-based image inpainting network, while HEAR-Net is a neural network for audio event recognition in environmental sounds.

As for the techniques of face recreation, the main difference they present concerning the last category is that they aim to control people's expressions in the images, which means that they can be generated by someone doing something that does not exist. While it is true that these techniques are somewhat older [263], they have been developing to a large extent in recent years thanks to the increase in the computing capacity of the systems, giving rise to tools such as Face2Face [253] achieving a performance of remarkable quality. The main problem presented by this technique is the non-guarantee of the coherence of the movements of the head since it only performs the migration of expressions.

All these changes again with the development of Deep Learning techniques where this inconsistency is corrected to some extent, adding recurrent neural networks or RNNs where features extracted from the audio are used to generate the

movements of the mouth [246]. For their part, [88] incorporated a new method for rendering the face using neural networks. In this sense, and to improve the existing techniques in state of the art, [138] proposed a novel method using a generative neural network, adding a Spatio-temporal architecture[2] that allows transforming renders of facial models into complete video output. With this contribution, a result characterized by a high degree of temporal space continuity is achieved. In addition, it is possible to migrate not only facial expressions but also the posture of the head, the control of the eyelids, and even the direction of the look.

2: Spatio-temporal architecture refers to a design framework that models and analyzes data or processes considering both spatial and temporal dimensions.

6.2 Ethical Challenges in the Fight Against Explicit Content

The use of Artificial Intelligence (AI) in detecting and managing adult content presents a range of ethical challenges that need to be carefully considered. AI systems, including machine learning algorithms and neural networks, are increasingly employed to filter and moderate online content. While these systems offer efficiency and scalability, they also raise several ethical concerns.

AI's opacity poses accountability issues, demanding transparency for public trust. Balancing privacy and preventing minors from accessing explicit content online necessitates comprehensive approaches involving technology, education, and parental control. It is important to address the effectiveness, biases, transparency, and broader societal impacts of AI in content moderation responsibly.

The complexity of AI algorithms often results in a lack of transparency[147], making it difficult to discern how decisions are made. This opacity can lead to a deficiency in accountability, particularly when errors occur. Ensuring that these AI systems operate in a transparent manner is crucial for public trust and responsible usage.

The challenge of preventing minors from accessing explicit content in the digital age is formidable. Traditional age verification methods have proven to be less effective online, prompting the exploration of AI-driven solutions. However, these methods also present privacy concerns[195] and the potential for intrusive surveillance. Early exposure to explicit content can adversely affect the psychological and emotional development of young people[205]. This concern necessitates a comprehensive approach that includes technology, education, and parental control to safeguard minors. Parents play an essential role in monitoring their children's online activities, and they need the tools and knowledge for effective oversight. Additionally, educational initiatives are important

for equipping young people with the skills to navigate the digital world safely and understand the risks associated with online content.

The ethical landscape surrounding the use of AI in content moderation and the challenges of limiting minors' access to explicit content are multifaceted. Addressing the effectiveness and potential biases of AI, maintaining transparency and accountability in these systems, and tackling the broader social impacts on minors are crucial for developing responsible and effective solutions.

6.3 The Role of AI

The detection of adult content using Artificial Intelligence (AI) has become an increasingly vital tool in moderating online media. Various studies and approaches have been employed to enhance the accuracy and efficiency of these AI systems.

One common approach is the use of machine learning models. These models are trained on large datasets containing examples of both adult and non-adult content. By analyzing these datasets, the AI learns to distinguish between different types of content based on various features such as visual cues and metadata. For instance, some studies have been conducted using convolutional neural networks (CNNs)[271, 89, 128], a type of deep learning model, to accurately classify images and videos. This method showed promising results in identifying explicit content with a high degree of accuracy.

Other approaches go beyond simple image recognition and include analysis of context, text and even the intention behind the content[223, 224]. For example, AI algorithms can be designed to understand the context in which an image is posted, which can be crucial in determining whether the content is explicit. Researchers at Stanford University[260] have developed algorithms that analyze not only the visual elements of the content but also the accompanying text and user comments to provide a more comprehensive assessment.

There are also a number of approaches using Deep neural network. One aproach for adult content detection is named ACORDE, proposed by J. Wehrmann et al. [272], employs CNNs to extract features and LSTMs to classify the outcome

without changing or retraining the CNNs. Even the least robust iteration of ACORDE, ACORDE-GN, performs far better than ACORDE.

Another tool, but not as widely used, is NLP (Natural Language Processing), that it is used to understand and interpret the textual content that often accompanies visual media. By analyzing the language used in descriptions, tags, and comments, AI can get better insights into the nature of the content. This method is particularly useful in filtering explicit content on social media platforms where textual information provides critical context.

6.4 Experimental Method in Content Filtering

As we have explained, the use of Neural Networks architectures have proven to be very effective for this type of tasks and that is why we have carried out a series of experiments to try to develop a new approach to adult content detection. In this section, we will go into the details of the method used in the experiment. In order to build the networks and analyze them, the open-source machine learning framework TensorFlow and the high-level neural networks API Keras were employed[3]. These widely-used tools offer a range of functionalities for developing and analyzing machine learning and deep learning models. Throughout this section, the different data-gathering techniques and data pre-processing that have been used are also explained.

3: TensorFlow is a deep learning framework that enables efficient numerical computations and machine learning models, while Keras is a high-level API simplifying model building.

6.4.1 Data Gathering & Pre-Processing

Given the broad scope and subjective nature of what constitutes 'adult content', we have established specific criteria in order to generate our dataset and use it for our experiment.

For the sake of this use case, we shall classify explicit images or other 'compromising' media as adult content, similar to the infamous 'dickpicks.' In contrast, a typical selfie will be categorized as 'Safe For Work' (SFW). Given these specific definitions, finding a suitable dataset that met our requirements was challenging.

To assemble a diverse collection of images that one might typically encounter on social media, we leveraged Reddit's API. This approach allowed us to create a dataset of 2000 photos, with an equal split of 50% 'Not Safe For Work' (NSFW) and 50% SFW content in both the training and testing sets. We sourced the NSFW photographs from subreddits like /r/GoneWild and /r/GayBrosGoneWild, while SFW images were gathered from /r/Selfies and /r/GayBrosGoneMild. The decision to include two subreddits catering to the gay community was driven by our observation of a lack of racial diversity in our dataset. This issue emerged because, as per Reddit's upvoting mechanism, the majority of photographs on more neutral subreddits are predominantly of women. This selection aimed to provide a more balanced representation of genders and sexual orientations in our dataset.

To create a dataset for identifying 'adult content', specific criteria were established, classifying explicit images as NSFW and typical selfies as SFW. Utilizing Reddit's API, a balanced dataset of 2000 images (50% NSFW, 50% SFW) was compiled from various subreddits. Image augmentation techniques were employed to prevent overfitting, and all images were resized to a uniform 224x224 pixels for model consistency.

To mitigate the risk of overfitting due to the relatively small size of our dataset, we employed image augmentation techniques. This process involves generating new images through various transformations such as translation, zooming, distortion, color scheme alteration, rotation, and flipping of the original images [180]. While augmentation helps in enhancing the robustness and generalizability of our model, it is worth noting that gathering a larger, more diverse dataset would be preferable [204].

Moreover, since all the images were sourced from a public website where users freely upload content of varying dimensions, uniformity was a challenge. To address this, we rescaled all images to a standard size of 224x224 pixels while maintaining their aspect ratio. This resizing was crucial to ensure consistency in input data for our model, thereby facilitating more accurate and reliable analysis.

6.4.2 Method Insight

In our experiment we carried out, Convolutional Neural Networks (CNNs) have been used, which are a type of neural network architecture designed specifically for processing data that has a grid-like topology, such as images or speech signals. The key characteristic of CNNs is the use of convolutional layers, which apply a set of filters to the input data, capturing local spatial correlations in the input data.

In a typical CNN, the convolutional layers are followed by one or more pooling layers[94], which reduce the dimensionality of the feature maps generated by the convolutional layers. This helps to reduce the number of parameters in the model and to make it more robust to variations in the input data.

CNNs have been shown to be highly effective in a wide range of applications[152], including image classification, object detection, and segmentation, as well as speech and natural language processing. By exploiting the spatial correlations in the input data, CNNs are able to learn complex features that are invariant to translation, rotation, and other spatial transformations, making them well-suited for many real-world problems.

In this experiment, Convolutional Neural Networks (CNNs), specifically the VGG16 architecture, were employed for Adult Content Filtering. CNNs, adept at processing grid-like data like images, use convolutional layers with filters to capture spatial correlations. VGG16, pre-trained on the ImageNet dataset, was fine-tuned for adult content detection using transfer learning. It utilizes pooling layers to reduce dimensionality and convolutional layers to learn complex features. The model, fine-tuned with binary cross-entropy loss and Adam optimizer, underwent evaluation on validation and test datasets. Training was conducted in batches over 10 epochs, with performance monitored to avoid overfitting.

Given that they perform convolution rather than matrix multiplication, convolutional neural networks (CNNs) have surpassed fully-connected neural networks as the most widely used Machine Learning approach for visual object detection (as in fully-connected neural networks) [118]. The number of weights is reduced as a result, and the network's complexity is reduced.

Furthermore, the images can be directly incorporated into the network as unprocessed inputs, thereby eliminating the requirement for feature extraction as in conventional learning algorithms. Convolutional Neural Networks (CNNs) have emerged as the first successful Deep Learning architecture due to their hierarchical layers, which progressively learn increasingly complex features through multiple layers that have been effectively trained[162].

Because our filter is an image classifier, it is perfect for CNN applications and was utilized in our example of adult content filtering.

In this experiment, a convolutional neural network (CNN) architecture called VGG16 is used for the task of Adult Content Filtering. The VGG16 architecture is a deep CNN that has achieved state-of-the-art performance on various computer vision tasks. The VGG16 model is initialized with pre-trained weights on the ImageNet dataset, which is a large-scale dataset containing millions of images from various categories. This pre-training allows the model to learn useful features that can be transferred to the Adult Content Filtering task.

In the case of Adult Content Filtering using the VGG16 architecture, transfer learning is used to initialize the model with pre-trained weights on the ImageNet dataset and fine-tune the model for the new task of Adult Content Filtering.

The first step is to load the pre-trained VGG16 model and remove the last layer specific to the ImageNet dataset. A new fully connected layer with a sigmoid activation function is added to output a probability score indicating the likelihood of the image containing adult content. The pre-trained layers are frozen to leverage the pre-trained weights and reduce the amount of training data needed. The model is fine-tuned on the dataset using binary cross-entropy loss and the Adam optimization algorithm to update the weights of the new fully connected layer and the last convolutional block. After fine-tuning, the model is evaluated on the validation and test datasets to measure its performance. Transfer learning with the VGG16 architecture allows for efficient training on the new task by leveraging the pre-trained weights on the ImageNet dataset and adapting the model to the new task through fine-tuning. The training is done in batches of 20 images for a total of 10 epochs. During each epoch, the model is evaluated on a validation dataset to monitor its performance and prevent overfitting.

6.4.3 Outcomes Overview

The experiment, leveraging deep learning techniques for Adult Content Filtering, utilized the VGG16 model as its backbone. The performance of the VGG16 model was evaluated based on three key metrics: the F1 score, the Area Under the Curve (AUC) for the Receiver Operating Characteristic (ROC), and overall accuracy. These metrics provide a comprehensive overview of the model's ability to correctly identify images containing adult content. The results were promising as show in Table 6.1, with the model achieving an F1 score of 0.85, an AUC of 0.94, and an accuracy of 0.88. These figures indicate that the VGG16 model exhibits a moderate to high level of effectiveness in classifying images according to their content.

The detailed analysis of performance metrics for the VGG16 model in adult content filtering shows that the model has a good balance between identifying relevant instances and

avoiding false alarms, as indicated by the F1 score of 0.85. However, there is room for improvement, especially in minimizing false negatives and false positives. The AUC value of 0.94 signifies a high degree of separability, implying that the model is capable of distinguishing between positive and negative classes with high reliability, reflecting its robustness against varying thresholds and general effectiveness in classification tasks. The accuracy score of 0.88 indicates a strong performance, with 88% of the images correctly classified. Nonetheless, given the sensitivity of the task, striving for higher accuracy is crucial to minimize the risk of inappropriate content slipping through the filters.

Table 6.1: Adult Content Experiment Results

Experiment	F1	AUC	Accuracy
Adult Content Filtering	0.86	0.94	0.88

The suboptimal performance of the model in this context may be due to several factors:

- Dataset Characteristics: The nature of the dataset, including the presence of images with ambiguous features, varied lighting conditions, or low resolution, could significantly challenge the model's classification capabilities. Such characteristics may lead to misclassifications, affecting the model's performance metrics.
- Model Architecture Limitations: While VGG16 is renowned for its proficiency in image recognition tasks, its architecture may not be ideally suited for this specific type of content filtering. The depth of VGG16 and its method of processing image features might not align perfectly with the nuances of detecting adult content, leading to potential shortcomings in its application.

To address these issues and enhance the model's performance, several strategies could be employed:

- In-depth Dataset Analysis: A thorough examination of the dataset and the model's performance on individual instances could unveil patterns in misclassifications. Identifying common characteristics among incorrectly classified images could guide adjustments in data preprocessing or suggest modifications to the model's architecture.

- Model Fine-tuning: Adjusting the VGG16 model's parameters, such as its layers, activation functions, or learning rate, could significantly impact its effectiveness. Fine-tuning allows the model to better adapt to the specific features of the dataset, potentially improving accuracy and the F1 score.
- Exploring Alternative Architectures: Considering different neural network architectures might yield better results. Architectures such as ResNet, Inception, or even custom models designed to address the specific challenges of adult content filtering could provide improved performance metrics.
- Advanced Data Augmentation: Implementing more sophisticated data augmentation techniques could enhance the model's ability to generalize from the training data and perform more effectively on unseen images. This might include variations in image quality, lighting, and occlusion to mimic the diversity of real-world scenarios more accurately.

By addressing these areas, the potential for significant improvements in model performance is substantial. Each strategy offers a pathway to not only refine the model's ability to classify content accurately but also to deepen our understanding of the challenges inherent in adult content filtering using deep learning techniques.

6.5 Research Summary and Future Research

Following the comprehensive research and experiments conducted, we have arrived at a nuanced understanding of the outcomes. While the results of our study might not be classified as exceptional, they certainly qualify as satisfactory. Importantly, these outcomes provide a valuable benchmark for future research in this field, offering insights and groundwork upon which subsequent studies can build.

The results, as detailed in the experiment results section, hint at several influential factors. A primary consideration is the nature of the dataset used in our study. It's possible that this dataset includes images that pose significant classification

The research provides a satisfactory benchmark for future studies in deep learning for image classification, despite not being exceptional. Key findings include challenges posed by the complex dataset and the suitability of the VGG16 architecture. Despite computational demands, deep learning proved efficient in analyzing complex data, highlighting potential and limitations for future advancements in this domain.

challenges. These might include images with complex features, ambiguous subjects, or varying quality, all of which could impact the model's ability to classify them accurately. Additionally, the choice of the VGG16 architecture for this task warrants scrutiny. While VGG16 has been a popular choice in various image recognition tasks, its suitability for this particular dataset and the specific requirements of this task may have been less than optimal.

The application of deep learning techniques in this domain underscores the need for substantial computational resources. The handling of large datasets, characteristic of deep learning applications, demands significant processing power. However, it's noteworthy that despite these demands, our research has demonstrated that the use of deep learning is not just feasible but also highly efficient in processing and analyzing complex data. This efficiency is crucial, as it suggests that even with the challenges presented by the dataset and architectural constraints, deep learning techniques can yield reliable and useful results.

While the results of our research might not redefine the field, they are indeed significant. They offer a solid foundation for future research, pointing out both the potential and the limitations of current methodologies and technologies in this area. This study serves as a stepping stone, paving the way for more refined, advanced research that could push the boundaries of what's currently achievable in the application of deep learning to complex image classification tasks.

Spam Analysis: LSTM Application

7

7.1 The Phenomenon of Spam

Spam, in the context of digital communication, refers to unsolicited and often irrelevant or inappropriate messages sent over the Internet, typically to large numbers of users, for the purposes of advertising, phishing, spreading malware, or other nefarious activities. Originally coined from a canned meat product, the term 'spam' has evolved to encompass a range of intrusive electronic communications. This section seeks to delineate the various dimensions of spam, categorizing it into subtypes such as email spam, social media spam, instant messaging spam, and others, each characterized by unique attributes and propagation methods.

Spam, unsolicited digital messages often for advertising or malicious activities, extends beyond email to social media and messaging. Originating from telegraphy, its evolution reflects digital communication's ethical and security challenges. This multifaceted phenomenon, evolving with technology, requires comprehensive understanding and advanced countermeasures.

The phenomenon of spam is multifaceted, encompassing not just the well-known email spam but also extending to unsolicited messages on social media platforms, unwanted SMS messages, and other digital communication channels. This breadth of coverage necessitates a robust and comprehensive definition that encapsulates all forms of digital spam, recognizing the shared characteristics that unify these various forms under a single conceptual umbrella.

The history and the increase of spam is intrinsically linked to the evolution of digital communication (Table 7.1). This section aims to trace the genesis of spam, from its early instances in telegraph communications to its proliferation in the era of the Internet. The journey of spam's evolution is not just a narrative of technological advancement but also a reflection of the evolving challenges in digital communication ethics and security.

Table 7.1: From 2019 to 2022, observed attacks escalated significantly, with a peak growth of 136.89% from 2019 to 2020, followed by steady increases, reflecting an alarming upward trend. [278].

Year	Nº of Attacks
2019	779,200
2020	1,845,814
2021	2,847,773
2022	4,744,699

Beginning with the first recognized instance of spam in 1864 via telegraph messages, the historical overview navigates through the advent of spam in email communications in the late 20th century, highlighting key milestones such as the first mass email sent by Gary Thuerk in 1978[276], often cited as the birth of modern email spam. The progression of spam parallels the development of the Internet itself, with each

new communication platform becoming a new frontier for spam activities.

The historical context sets the stage for understanding the current complexities and challenges associated with spam. By exploring how spam has adapted to various communication mediums over time, the section underscores the persistent and ever-evolving nature of spam, thereby establishing the foundational knowledge necessary to appreciate the subsequent discussion on current trends, impacts, and the role of advanced technologies like LSTM in combating this persistent digital challenge.

7.2 Analyzing the Landscape of Spam

In this section, we take a closer look at the current state of spam, using recent statistics to paint a picture of its prevalence and evolution. The digital landscape is constantly shifting, and with it, the nature of spam. By examining the volume of spam across different platforms, such as email, social media, and mobile messaging, we gain insight into the patterns and techniques employed by spammers. This analysis is supported by data from leading cybersecurity firms and research institutions, which track spam trends and report on their findings annually.

The current state of spam, characterized by its evolving sophistication and shifting content, poses significant challenges. Increasingly mimicking legitimate communication, spam now includes politically motivated content and phishing attempts, impacting individuals, organizations, and internet infrastructure. This necessitates advanced detection methods and has influenced legal frameworks worldwide.

A notable trend in the realm of spam is the increasing sophistication of spam messages and techniques. Gone are the days when spam was easily identifiable by its poorly written content and blatant advertising. Modern spam often mimics legitimate communication[209], making it more challenging to detect. This evolution calls for advanced detection and filtering methods, as traditional approaches become less effective.

The shift in spam content is also noteworthy[123]. While commercial advertising still constitutes a significant portion of spam, there has been a rise in politically motivated spam and phishing attempts aimed at data theft and fraud. This change in content reflects broader societal and technological shifts, indicating the adaptive nature of spammers to current events and technological vulnerabilities.

The impact of spam extends beyond mere annoyance; it has significant implications for individuals, organizations, and

society as a whole. At the individual level, spam can lead to information overload, reducing productivity and potentially exposing users to harmful content or scams[1]. For organizations, spam represents a security threat, as it is often a vector for malware and phishing attacks. The financial implications are considerable, with businesses investing heavily in spam filtering technologies and suffering losses due to successful spam attacks.

1: Scams are fraudulent schemes aiming to deceive, manipulate, or defraud individuals or entities by using deceitful tactics for personal gain.

On a broader scale, spam affects the very infrastructure of the Internet. The sheer volume of spam traffic can strain network resources, impacting service quality. Furthermore, the battle against spam influences legal and regulatory frameworks worldwide, as governments seek to protect consumers and businesses from spam-related threats. This has led to the enactment of various anti-spam laws and regulations, though their effectiveness varies by region and enforcement capacity.

In summary, the current landscape of spam is characterized by its vast scale, evolving tactics, and significant impact on individuals, organizations, and technological infrastructures. Understanding these dynamics is crucial for developing effective strategies to combat spam, which will be further explored in the context of LSTM applications in subsequent sections of this thesis.

7.3 Evolution of Spam Detection Methodologies

In the fight against the ever-evolving phenomenon of spam, various detection methodologies have been developed and implemented over the years. This section provides an overview of these traditional spam detection techniques, setting the foundation for understanding their evolution.

Initially, spam detection was predominantly rule-based, relying on specific criteria set by administrators[279]. These rules could include keyword filtering, where emails containing certain words were marked as spam, or blacklists, which blocked messages from known spam sources. Rule-based systems, while effective initially, required constant updating to keep up with the cunning adaptations of spammers.

Another conventional method involves heuristic-based filtering[108, 222]. Heuristic filters analyze the characteristics of an email, such as header content and email structure, to assign spam probability scores. These filters are more sophisticated than rule-based systems, as they learn from identified spam characteristics but still rely on predefined algorithms and patterns.

Traditional spam detection methods, including rule-based, heuristic, and content-based filtering, have evolved over time. Initially effective, these methods struggle with modern spam's cunning adaptations, like image-based spam, leading to high false positives and negatives. Their resource-intensive nature underscores the need for advanced techniques like LSTM networks.

Content-based filtering, another prevalent approach, uses the statistical probability of words in legitimate versus spam emails to make its determinations. Techniques like Bayesian filtering fall under this category. These filters are dynamic, learning and adapting from user feedback on what constitutes spam for them[8, 99, 225].

While these traditional methods have been instrumental in combating spam, they possess inherent limitations, especially when faced with modern spamming techniques.

One significant limitation is the reliance on predefined rules or patterns. Spammers continuously innovate, crafting messages that evade these rules. For instance, image-based spam, where text is embedded in images, can bypass text-based filters.

Another challenge is the dynamic nature of spam. As spam evolves, the static nature of traditional filters often leads to a high rate of false positives (legitimate messages marked as spam) and false negatives (spam messages not identified). This inaccuracy can be problematic, either blocking important communications or allowing harmful content through.

Furthermore, traditional methods are often resource-intensive. Maintaining and updating spam filters require considerable time and effort, especially in large-scale or enterprise settings[2]. This is compounded by the vast volume of spam, which necessitates powerful and efficient processing capabilities.

2: Spam is estimated to cost businesses about $20.5 billion each year, including decreased productivity and technical expenses[254].

While traditional spam detection methods laid the groundwork for spam filtering, their limitations in the face of sophisticated and evolving spam tactics highlight the need for more advanced techniques. This sets the stage for exploring the application of LSTM (Long Short-Term Memory) networks in spam detection, which promises a more dynamic and effective approach in tackling modern spam challenges.

7.4 Applying LSTM to Spam Detection

After a review of all previous approaches and studies and an understanding of the difficulties and needs of this task, it has been chosen to employ Long Short-Term Memory (LSTM) networks for spam detection, as it is anchored on several key attributes of LSTM that make it particularly suitable for this task.

Employing Long Short-Term Memory (LSTM) networks for spam detection addresses key challenges due to their proficiency in processing sequential data, adaptability to evolving spam tactics, nuanced language understanding, and scalability for large datasets.

Firstly, the nature of spam and legitimate messages often involves sequences of words and phrases where context and order are crucial. LSTM, a type of recurrent neural network (RNN), excels in processing and making predictions based on such sequential data. Unlike traditional methods that treat words or phrases independently, LSTM can understand and remember context over long sequences, providing a deeper comprehension of the content[285]. This ability is paramount in distinguishing sophisticated spam that mimics legitimate messages.

Secondly, the evolving nature of spam presents a challenge that demands an adaptive and dynamic approach[238]. LSTM networks are known for their ability to learn and adapt over time. By training on a continuously updating dataset, an LSTM model can adjust to the ever-changing patterns of spam, making it more robust against novel or sophisticated spam tactics that traditional filters might miss.

Another significant advantage of LSTM in spam detection is its capability to handle the nuances of language. Spam often contains subtle cues and variations that are difficult for rule-based systems to catch[247]. LSTMs, through their learning mechanisms, can understand these subtleties and effectively differentiate between spam and legitimate messages, reducing the rate of false positives and negatives.

The scalability and efficiency of LSTM models make them suitable for handling the large volumes of data typical in spam filtering scenarios. With the increasing amount of digital communication, a system that can efficiently process and learn from large datasets is indispensable. LSTMs, with their efficient training and inference capabilities, meet this requirement.

7.4.1 Methodology and Experiment Design

In our proposed methodlogy, Long Short-Term Memory (LSTM) have been used, which is a type of recurrent neural network (RNN) architecture that is designed to address the problem of vanishing gradients during backpropagation through time. Unlike traditional RNNs, which use a simple recurrent layer to store information about past inputs, an LSTM includes a memory cell and a set of gates that control the flow of information into and out of the cell (Figure 7.1). The memory cell can maintain information over long periods of time, while the gates allow the model to selectively remember or forget information based on the current input and the contents of the cell. This makes LSTM well-suited to tasks that require the model to maintain context and remember important details from earlier inputs, such as speech recognition, language translation, and video analysis.

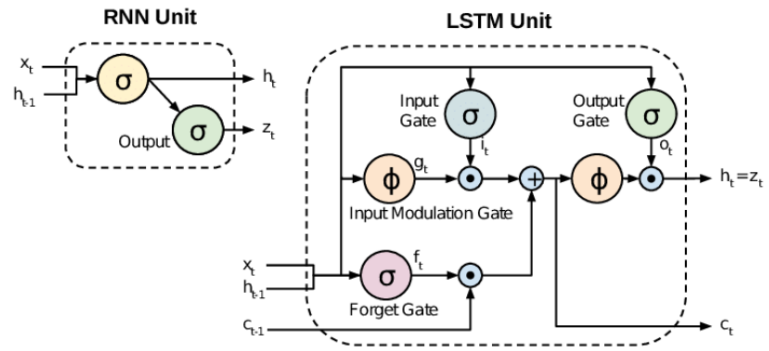


Figure 7.1: Comparison between RNN and LSTM architecture. [258].

In theory, Recurrent Networks [131] can use their feedback connections to store activations as representations of recent input events ("short-term memory," as opposed to "long-term memory" embodied by steadily changing weights). Many applications, such as speech processing, non-Markovian control, and music composition, could benefit [116].

LSTMs are made to prevent the problem of long-term dependency. They have a chain-like structure and can remember information for long periods [104], but the repeating module has a distinct design. There are four neural network layers instead of just one as shown in Figure Figure 7.1.

We use them to detect SPAM because of their ability to remember information, which is extremely useful for semantic relation and sentiment analysis [267].

In order to implement an LSTM model in our experiment, the first step is to define the architecture of the model. The LSTM model typically consists of several layers, including an input layer, an embedding layer, an LSTM layer, a dense output layer, and an output activation function. The input layer of the model receives the preprocessed data, which is then fed to the embedding layer. The embedding layer converts the integer-encoded words into dense vectors of fixed size, which can then be processed by the LSTM layer. The LSTM layer is the core of the model and is responsible for learning the patterns in the input data. It is made up of several memory cells that can store information over time and gates that regulate the flow of information through the cells. The number of units in the LSTM layer should be chosen based on the complexity of the problem and the size of the dataset. The output of the LSTM layer is then passed to a dense output layer, which applies a linear transformation to the output of the LSTM layer. The output layer has a single unit for binary classification (spam or ham). Finally, the output of the output layer is passed through Sigmoid activation function, which maps the output to a probability distribution over the possible classes.

We used Long Short-Term Memory (LSTM) networks, a type of RNN architecture designed to overcome vanishing gradients. Their unique structure enables prolonged information retention and selective memory usage. The LSTM model, with its complex multi-layered architecture, including dropout and Adam optimizer, is tailored for semantic analysis and sentiment detection, crucial for effective spam identification. It's trained over 10 epochs with a learning rate of 0.001, ensuring balanced convergence and stability.

Between each layer dropout of 0.8 has been added. Dropout is a regularization technique used to prevent overfitting in neural networks. In simple terms, dropout involves randomly dropping out (i.e., setting to zero) some neurons in a neural network during training. As in the previous experiment, the model has been compiled with Adam optimizer. The binary_crossentropy function, shown in Equation 7.1, has been used as loss function, where y is the true label (0 or 1) and p is the predicted probability of the positive class (i.e., the probability that the sample belongs to the positive class).

$$-(y * log(p) + (1 - y) * log(1 - p)) \tag{7.1}$$

As a final step, the training has been done in a total number of 10 epochs where an epoch represents a complete iteration through the dataset during training, and a batch size of 20, which refers to the number of samples processed before updating the model weights. Learning rate determines the step size at each iteration while moving toward a minimum of a loss function, and we use a learning rate of 0.001 which

is a balanced value between fast convergence and stability.

Data Gathering & Pre-Processing

For this study, we choose to look at SMS SPAM independently from email SPAM. The major cause of this is that text messages, unlike emails, are short and include less statistically differentiable information. As a result, there are fewer characteristics available to detect spam SMS because of this. Email spam filtering algorithms fail in the case of SMS because informal languages such as regional terminology, idioms, phrases, and acronyms have a significant impact on text messages. The dataset was collected from Kaggle[3][181]. We utilized a dataset of over 5000 messages, 4457 of which were used for training, and the other SMS were used for validation. In both sets, 13% of the labels are dispersed for SPAM, and 87% are spread for HAM.

3: Kaggle is an online platform for data science and machine learning competitions, collaboration, and datasets, fostering innovation in the field.

First, we tokenized text messages which turns them into sequences, in order to detect spam. This is a common Natural Language pre-processing task. Each word in the lexicon has a vector of similarity once the sequences have been converted into word embeddings using word2vec[4], which makes it easier for the network to gain semantic meaning using LSTMs.

4: Word2Vec is a technique that transforms words into numerical vectors, capturing semantic relationships for various NLP tasks.

7.4.2 Results and Analysis

Before presenting the detailed results, it's crucial to briefly recap the methodology. This study employed Long Short-Term Memory (LSTM) networks, a type of recurrent neural network (RNN) specialized in capturing long-term dependencies, to detect SMS spam. The dataset comprised a mix of genuine and spam messages, which were preprocessed to fit the input requirements of the LSTM model. The model was trained with a specific architecture designed to optimize performance for the spam detection task, including layers tailored for sequence processing and classification.

The LSTM model achieved an Accuracy of 0.96, indicating that 96% of the SMS messages were correctly classified as spam or non-spam. This high accuracy demonstrates the model's effectiveness in distinguishing between the two classes.

The Area Under the Receiver Operating Characteristic curve (AUC) reached 0.99%, showcasing the model's excellent capability to differentiate between spam and non-spam messages across various threshold settings. AUC is a critical metric for classification problems, especially in imbalanced datasets, as it provides a comprehensive measure of the model's performance at all classification thresholds.

The F1 score, a harmonic mean of precision and recall, was 0.97%. This score signifies not only the model's precision in identifying spam messages but also its ability to minimize false positives and false negatives. A high F1 score is particularly important in spam detection, where both overlooking actual spam and misclassifying legitimate messages can have negative consequences.

The experiment of utilizing deep learning techniques for detecting SMS spam has shown promising results, indicating that the application of LSTM algorithms is a viable method for this purpose. The accuracy, AUC, and F1 score metrics obtained in this experiment have all demonstrated high levels of performance, with the model correctly classifying spam and non-spam messages with precision and high discriminatory power. The F1 score, which considers both precision and recall, has shown that the model can correctly identify spam messages while minimizing the number of false positives. These results indicate that the deep learning approach utilized in this study has significant potential for improving the detection of SMS spam, with implications for enhancing the efficiency of spam filtering systems.

Experiment	F1	AUC	Accuracy
Spam Filtering	0.97	0.99	0.96

Table 7.2: Spam Filtering Experiment Results

The LSTM model's high performance can be attributed to its ability to capture and utilize the sequential nature of text data, a crucial aspect of SMS spam detection. The model's architecture, including the number of LSTM units and the configuration of fully connected layers, was optimized through experimentation to achieve the best possible results.

The success of the LSTM model in detecting SMS spam has several implications. Firstly, it validates the efficacy of deep learning techniques, particularly RNNs, in text classification

tasks. Secondly, the high performance achieved suggests that LSTM-based models could be practically implemented in spam filtering systems, potentially improving their efficiency and accuracy.

Future work could explore several avenues:

- Model Optimization: Further tuning the model's architecture and hyperparameters could enhance performance. Additionally, exploring other RNN architectures or hybrid models could yield improvements.
- Feature Engineering: Incorporating more sophisticated feature extraction techniques, such as word embeddings or sentiment analysis, might improve the model's understanding of the text's context and nuances.
- Real-world Testing: Deploying the model in a real-world setting would provide insights into its practical performance and operational challenges, including handling evolving spam tactics.
- Broader Application: Investigating the model's effectiveness in other domains of text classification, such as email spam detection or social media moderation, could expand its applicability.

Nevertheless, we see this as a success since our experiment shows that powerful models can be developed using deep learning techniques without requiring significant adjustments.

7.5 Synthesizing Insights on LSTM's Role in Advancing Spam Detection

Throughout this experiment, we have embarked on a comprehensive journey exploring the multifaceted landscape of spam and the pivotal role of Long Short-Term Memory (LSTM) networks in advancing spam detection. The journey began with an elucidation of what constitutes spam, its historical evolution, and its impact on digital communication and society. This provided a foundational understanding of the complexity and dynamism inherent in spam detection.

The core of this methology focused on the theoretical underpinnings of LSTM networks and their suitability for spam

detection. LSTMs are adept at handling sequential data, offering a deep understanding of context and dependencies in language, which is crucial for effective spam detection. The adaptability and learning capability of LSTMs, essential in keeping pace with the evolving nature of spam, were emphasized as key motivators for their selection in this research.

The experiment demonstrated the feasibility of LSTM in accurately identifying spam. These findings reinforce the premise that LSTM networks can offer significant improvements in spam detection, striking a balance between accuracy, adaptability, and efficiency.

This comprehensive study explores spam's evolution and the effectiveness of Long Short-Term Memory (LSTM) networks in detecting spam. Focusing on LSTM's ability to understand language context and adapt to spam's changing nature, the research demonstrates LSTM's potential in accurately identifying spam. The study highlights the necessity and challenges of integrating advanced AI like LSTM in spam detection, underscoring its significance in evolving digital communication.

Looking ahead, the role of LSTM in spam detection appears not only promising but also necessary. As digital communication continues to expand and evolve, so too will the nature and complexity of spam. The adaptability and learning capabilities of LSTM networks position them as a robust solution to this ever-growing challenge. However, it's important to recognize that the field of artificial intelligence and machine learning is continually advancing. The future may unveil more sophisticated models or approaches that could further enhance spam detection capabilities.

Additionally, the integration of LSTM into existing spam detection frameworks must be approached with a nuanced understanding of its limitations and challenges, such as computational requirements and the need for extensive training data. Ongoing research and development, coupled with practical implementations, will be crucial in addressing these challenges and optimizing LSTM's effectiveness in real-world scenarios.

The exploration of LSTM in the realm of spam detection presents a compelling narrative of the potential of advanced AI techniques to revolutionize traditional methodologies. The insights gleaned from this thesis not only contribute to the academic discourse but also pave the way for practical advancements in combating spam. As we continue to navigate the digital age, the role of LSTM in spam detection is poised to be a cornerstone in the ongoing effort to secure and streamline digital communication channels.

8 Conclusions

In the era of digital transformation, cyber security is at the centre of the landscape of technological progress and emerging threats, becoming a fundamental element in safeguarding the socio-economic structures of our society. As digital technologies penetrate critical sectors such as finance, healthcare and governance, there is an increasing need to strengthen cybersecurity techniques.

As cybersecurity risks and dangers have evolved and increased, the scientific field has placed greater emphasis on cybersecurity studies, merging classical and cutting-edge approaches. This thesis focuses on the movement towards incorporating artificial intelligence (AI) into cybersecurity tactics. It details the preliminary steps taken to leverage AI capabilities in the detection, mitigation and prevention of cyber threats, representing a substantial shift in the subject.

The goal of this thesis is harnessing artificial intelligence to offer new methodologies and techniques for the detection of cyberthreats. It proposes new advanced AI methods to address 4 main cybersecurity issues with the greatest impact today. Through the experiments, this thesis makes significant contributions to the field of cybersecurity, by introducing new methodologies and applying them to real environments, in the face of new emerging cyberthreats. The results of the experiments demonstrate both the viability and efficiency of these methods, as well as the improvement of existing methods.

8.1 Summary of Research Objectives

This thesis pivots on the following fundamental hypothesis that underline the potential transformation AI can bring into cybersecurity:

"The application of artificial intelligence in cybersecurity significantly enhances threat detection and response, enhancing traditional systems due to its real-time data processing, adaptability

to emerging threats, and capability to proactively predict and neutralize attacks."

The main objective focused on exploring AI's impact and applications within cybersecurity, identifying its advantages, limitations, and opportunities for advancement. To achieve this, the research was structured around several specific objectives:

- Conduct a comprehensive literature review to understand AI's integration into cybersecurity.
- Evaluate the effectiveness of current AI applications in this domain.
- Present original research on AI's application to specific cybersecurity challenges.
- Anticipate future cyber threats and AI's role in addressing them.
- Offer recommendations for organizations to integrate AI into their cybersecurity strategies effectively.

With this starting point, we initiated the research in two separate stages. In the first, the objective was to understand in detail current cybersecurity issues and risks, to understand the latest developments in the field of artificial intelligence and what approaches have been taken to align these two main areas. Subsequently, once the current problems were identified, we proposed methodologies and techniques that mitigate these selected problems, reducing detection costs and facilitating automation.

8.2 Contribution to the Literature

This doctoral thesis has resulted in three main contributions, which advance the existing state of the art in the field of AI and Cybersecurity.

In the first contribution we have focused on the study of artificial intelligence techniques in the most relevant cybersecurity issues. Firstly, we look at the state of the art, investigating existing methodologies and approaches, giving an overview of the landscape. After investigating these aspects, we work on a series of experiments that try to show both the advantages and disadvantages of applying artificial intelligence. The research gives a practical approach to employing artificial intelligence, particularly deep learning techniques, to

tackle three primary cybersecurity challenges: spam filtering, malware detection in portable executables (PEs), and adult content filtering. The methodology involves using Long Short-Term Memory (LSTMs) networks for spam detection, Deep Neural Networks (DNNs) for malware identification, and Convolutional Neural Networks (CNNs) combined with transfer learning for distinguishing adult content. The results demonstrated high efficacy, showcasing the practical applicability and effectiveness of deep learning in enhancing cybersecurity measures without complex system designs. This first contribution has been splited into the Chapter 7, Chapter 6 and Chapter 5, as the experiments in this contribution have been developed independently.

Secondly, to go into more detail on one of today's most relevant problems, malware on mobile devices, a new malware detection methodology based on the analysis of application bytecode has been developed. This approach leverages Convolutional Neural Networks (CNN) to analyze the graphical representations of Android application bytecodes, utilizing models like VGG16, RESNET50, and InceptionV3. The methodology emphasizes the transformation of bytecodes into images, enabling the CNNs to effectively learn and detect malware patterns. The research confirms the potential of using CNN for Android malware detection and emphasizes the importance of graphical representation of bytecodes in preserving information during the conversion process. Results from the study highlight the system's high accuracy and efficiency in identifying malware, presenting a cost-effective solution for early malware detection and enhancing mobile security.

Thirdly, in order to address other areas in which security plays a very important role nowadays, importance has been given to network traffic security. The needs and current situation of these environments have been studied, and an adaptive methodology has been designed. This methodology has been validated in a real environment, in order to provide a more realistic and practical approach. The research focuses on the necessity of enhancing network traffic security, particularly through the detection of malicious activities in sampled NetFlow traffic. The methodology involves three main steps: data gathering from NetFlow flows, converting this data into 2D images, and then training a CNN model with these images. The results demonstrated a high accuracy,

highlighting the effectiveness of applying artificial intelligence in identifying network attacks within sampled traffic, although accuracy drops significantly at higher sampling rates, suggesting limitations in the method's scalability.

Along with the direct contributions that this thesis work offers, it is also necessary to understand the context and secondary contributions. In the current era, alongside the surge in Artificial Intelligence (AI), there has been an increase in the number of experts and researchers in the field. Consequently, the volume of research applying AI to cybersecurity has also grown. It is notable that existing research in this domain often builds upon a comprehensive understanding of AI, enabling more in-depth investigation into applied methodologies and technologies. As a cybersecurity expert, my focus has been on approaching this from a cybersecurity-centric perspective, prioritizing the existing needs within the cybersecurity landscape and addressing them as required.

The research carried out throughout this thesis focuses on analysing and addressing the needs of cybersecurity. Furthermore, we have tried to focus these experiments on both studying and comparing existing methodologies and approaches, as well as developing new efficient and successful methodologies both from an academic approach and taking into account real and business environments. The results obtained have practically all been favourable, and in those cases that have not been exceptional, work has been done to understand and explain the reasons and causes.

8.3 Research Findings and Limitations of the Research

Having outlined the methods developed in the course of the doctoral research, let us analyse the results obtained and the limitations we encountered.

The experiments carried out in the first contribution of this doctoral thesis have been divided into the Chapter 7, Chapter 6 and Chapter 5. These three experiments entitled "Spam Analysis: LSTM Application", "Content Filtering: Adult Imagery" and "PE Malware Analysis: DNN Exploration" respectively, address in different ways the problems of SPAM

filtering, malware detection in portable executables (PEs), and adult content filtering.

In tackling the issue of unsolicited SMS detection, the research utilized LSTM-based models that showed high efficacy in classifying messages as spam or non-spam, showcasing the practicality of employing LSTM algorithms for spam detection without the need for extensive model adjustments. This reinforces the viability of deep learning approaches in combating spam effectively.

In the realm of portable executables detection, the study achieved promising outcomes, demonstrating the model's capability to accurately distinguish between malware and benign software with high accuracy, AUC, and F1 score. This highlights the potential of deep learning in bolstering the security of computing systems and networks, despite the complexities posed by the varied nature of malware samples and the omission of malware subcategories in the classification process.

The experiment on adult content filtering, which employed the VGG16 model, also yielded moderate to high levels of accuracy, F1 score, and AUC. However, this area of the study suggested that there is potential for improvement, possibly due to challenges related to the dataset or limitations inherent in the VGG16 architecture for this specific application. This opens avenues for future research to explore model optimization or the use of alternative architectures to enhance the effectiveness of adult content filtering.

These findings collectively underscore the substantial promise of deep learning methodologies in addressing critical cybersecurity challenges, offering insights into their practical applications and potential areas for further refinement and exploration.

While the experiments demonstrated the effectiveness of deep learning in addressing cybersecurity issues, several limitations were identified:

- Dataset Quality and Size: The quality and comprehensiveness of training data are crucial for model performance. Inadequate or unrepresentative datasets can hinder the model's ability to generalize and detect new threats or malware mutations.

- Model Robustness: Ensuring the model's robustness against adversarial attacks is vital. Attackers continually evolve their methods, potentially compromising the effectiveness of existing models and avoidance techniques.
- Explainability and Interpretability: Understanding how deep learning models make decisions is essential for trust and adoption. The "black box" nature of these models can be a barrier to their broader acceptance in critical applications like cybersecurity.
- Computational Resources and Expertise: Implementing deep learning solutions requires substantial computational resources and technical knowledge, which may not be readily available in all organizations.

The experiment entitled "*Malware Detection: Neural Insights*" in Chapter 3, which was conducted on Android malware detection using Convolutional Neural Networks (CNN) including VGG16, RESNET50, and InceptionV3, showcased significant findings in the field of cybersecurity. Through the application of these models on a dataset comprising 13,000 applications, the study demonstrated an impressive capability in identifying malicious software with accuracies reaching up to 99%. Particularly, the VGG16 model outshined others in terms of precision, recall, and F1 score, indicating a robust balance between the sensitivity and specificity of malware detection. The research underscored the efficacy of converting DEX files into image representations for CNN processing, which preserved the integrity of the bytecode information, ensuring no loss of critical data during the transformation process.

On the other hand, despite the promising results, the research acknowledged certain limitations. One of the primary constraints was the focus on a specific dataset and the potential for varying results when applied to other datasets with different characteristics or newer malware samples. The study's reliance on pre-trained models also raises questions about the adaptability and scalability of the approach to evolving malware threats that continuously adapt to bypass detection mechanisms. Moreover, the research pointed out the need for further investigation into the explainability of the CNN models used. Understanding the decision-making process of these models is crucial for improving their reliability and trustworthiness in practical applications. Future work

is intended to further explore these areas in depth, exploring the intricacies of model decision-making and extending the methodology to a wider array of malware categories, thereby enhancing the model's applicability and effectiveness in real-world scenarios.

The research and experiment carried out in Chapter 4, called "*NetFlow Defense: CNN Surveillance*" demonstrated the potential of Convolutional Neural Networks (CNNs) in detecting malicious traffic within sampled NetFlow data. In this experiment, our methodology achieved up to 94.15% accuracy at a sampling rate of 500. However, the study also highlighted significant limitations, particularly the sharp decline in accuracy at higher sampling rates, such as 1000, where it fell to around 50%. This drop underscores the challenges in applying this approach in real-world environments with higher sampling rates, indicating a need for further research and development to enhance scalability and effectiveness in diverse network conditions.

8.4 Future Research Directions

This doctoral thesis represents the initial phase of an ambitious and extended research endeavor aimed at exploring and harnessing the potential of deep learning techniques within the realm of cybersecurity. It sets the foundation for a comprehensive and long-term investigation, dedicated to advancing our understanding and capabilities in addressing some of the most pressing cybersecurity challenges of our time. As we get deeper into the complex intersection between artificial intelligence and cybersecurity, this work serves as a cornerstone, paving the way for future explorations and innovations. It is a step on the path towards developing more robust, efficient, and intelligent cybersecurity solutions that can adapt to and counteract the ever-evolving landscape of cyber threats. In this context, the research presented in this thesis is not an endpoint but a beginning, marking the inception of a broader scholarly pursuit that aims to contribute significantly to the field of cybersecurity through the lens of deep learning technologies.

On the one hand, defining the future lines of the first experiment, it is worth mentioning the potential of deep learning in addressing cybersecurity challenges, including malware

detection, spam filtering, and adult content filtering. It showcase that deep learning models can achieve high performance metrics, such as accuracy, AUC, and F1 scores, in these domains. However, they also acknowledge the limitations and challenges that accompany the application of deep learning in cybersecurity.

The findings suggests that further research and experimentation are vital to refine the performance of deep learning models in cybersecurity tasks and to expand their application to a broader range of security issues. This ongoing development is crucial to ensure that deep learning techniques remain effective against the continuously advancing landscape of cyber threats. This experiments advocate for a cautious and informed approach to applying deep learning in cybersecurity, highlighting the importance of continued innovation and adaptation in the field.

For malware detection, the study highlights the challenge of dealing with a wide variety of malware samples without differentiating between subcategories, suggesting that future work could focus on refining models to recognize and categorize different types of malware more accurately. This could involve developing more complex models or incorporating additional features that capture the unique characteristics of various malware types.

In spam detection, the success of LSTM models indicates a potential for further exploration in natural language processing applications within cybersecurity. Future research could aim to enhance the models' ability to understand the nuances of language, including slang, idioms, and emerging terminologies, to maintain high accuracy in spam detection amidst evolving spamming techniques.

The adult content filtering experiment, despite its successes, points to the need for improvement, particularly in dealing with the challenges posed by diverse and complex datasets. Future efforts could explore alternative deep learning architectures or advanced image processing techniques to improve the classification accuracy of such content. Additionally, expanding the dataset to include a broader range of content and contexts could help in developing more robust and generalizable models.

Overall, this experiments suggests a continuous need for refining deep learning models to keep pace with the evolving

landscape of cybersecurity threats. This includes addressing challenges related to the quality and size of training datasets, enhancing the robustness of models against adversarial attacks, and improving the explainability and interpretability of deep learning systems in cybersecurity applications. The research underscores the importance of further experimentation and development to extend the applicability of deep learning across a wider array of cybersecurity issues, ensuring these models remain effective and relevant in combating emerging threats.

For future work, It is planed to further investigate the explainability of the Convolutional Neural Networks (CNNs) used in the study, aiming to gain a deeper understanding of the decision-making processes behind the models. This endeavor is critical for improving the ability to classify and understand new malware mutations, particularly in the face of challenges such as code obfuscation.

Furthermore, It is planed to extend the evaluation of the proposed detection methodology across various malware subcategories. This expansion is expected to refine the ability of the system to accurately classify and categorize malware samples, thereby offering a more nuanced understanding of the malware landscape. Such advancements will not only contribute to the technical depth of malware detection methodologies but also enhance their practical applicability in securing Android devices against a wide array of threats.

By addressing these issues, I seek to solidify the foundations laid by the current study and push the boundaries of what can be achieved in the field of Android malware detection using deep learning techniques, improving the image generation and bytecode transformation methodology to optimise detection rates as much as possible.

Finally, the experiment focused on the detection of malicious traffic, outlines two main avenues for future work. On the one hand, more research and testing needs to be done using real production environments, since, as mentioned above, this experiment has focused on real traffic but using a dump of these flows, and not dealing with the volume of data and the performance load that could be caused. On the other hand, the impact of different NetFlow features on model performance needs to be explored, which could lead to the investigation of different CNN architectures or training

methods. Additionally, optimizing the image creation process for real-environment implementation could be a focus, given its critical role in the methodology.

8.5 Overall Conclusions

The speed at which technology is advancing in today's age is unprecedented. Just a few decades ago, the concept of artificial intelligence was considered science fiction. Today, this area is not only part of our reality, but fundamental to the evolution and sustainable development of our digital environment. As we advance along this technological path, we face complex challenges that require innovative and proactive solutions. This thesis has explored in depth how the application of artificial intelligence can revolutionise the field of cybersecurity, offering more sophisticated and effective tools for threat detection and prevention in a constantly evolving digital landscape.

With technological advancements, we increasingly rely on tools embedded with artificial intelligence for daily tasks, ranging from recommendation engines to software that assists with routine chores. It is crucial to also consider the risks and dangers associated with this rapid development and deployment of such technologies.

When discussing artificial intelligence, it is essential to address the apprehension among untrained users towards AI, often due to the "black box" nature of these systems. This lack of transparency can lead to fear, mistrust, and social unrest, not only among lay users but also within the scientific community. There is a degree of uncertainty, stemming from the lack of a detailed understanding of the inner workings of the technology.

While there is a vast body of research applying artificial intelligence, most studies are confined to specific fields and involve comparative analyses. A significant portion of the community lacks a deep understanding of the foundational principles and mechanisms of AI, perpetuating the notion of AI as a "black box".

On the other hand, cybersecurity remains one of the most pertinent topics in recent years. Unlike AI, it has not experienced a sudden explosion in popularity but has consistently

maintained its significance and necessity. This thesis aims to bridge these two crucial areas, leveraging AI technology to address one of the most pressing needs in the field of cybersecurity.

This thesis has successfully demonstrated the pivotal role of artificial intelligence (AI) in revolutionizing cybersecurity, particularly in the domain of cyber threat hunting. Through meticulous research and a series of experiments, it has been established that AI not only enhances the efficiency and effectiveness of identifying, mitigating, and preventing cyber threats but also significantly contributes to the evolution of cybersecurity methodologies from reactive to proactive and predictive strategies.

The findings points to the transformative potential of AI in cybersecurity, highlighting its capability in rapid threat detection, real-time incident response, and addressing the cybersecurity skills gap. The integration of AI technologies like machine learning, deep learning, and neural networks has shown promising results in malware detection, network security, content filtering, and spam analysis, underscoring the versatility and adaptability of AI in tackling diverse cybersecurity challenges.

The research also sheds light on the limitations and ethical considerations inherent in the deployment of AI in cybersecurity, advocating for a balanced approach that considers the implications of AI technologies on privacy, data integrity, and the potential for misuse.

The process of developing this thesis has been characterized by constant review of the state of the art in this field. Almost on a daily basis, studies and advances are published, demonstrating progress or new approaches, which has involved a certain cyclical nature in the review when analyzing technologies and conducting experiments.

This thesis not only contributes significantly to the existing body of knowledge in cybersecurity and AI but also lays a robust foundation for future innovations in this rapidly evolving field. It offers a point of view both by analysing the theoretical framework and by offering a practical approach thanks to the experiments carried out. The findings of this research advocate for a more integrated, AI-driven approach

to cybersecurity, promising a more resilient digital infrastructure capable of countering the sophisticated cyber threats of the modern era.

Bibliography

[1] Shubair Abdulla et al. 'Article: Setting a Worm Attack Warning by using Machine Learning to Classify NetFlow Data'. In: *International Journal of Computer Applications* 36 (Oct. 2011), pp. 49–56 (cited on page 109).

[2] Erwin Adi, Zubair Baig, and Philip Hingston. 'Stealthy Denial of Service (DoS) attack modelling and detection for HTTP/2 services'. In: *Journal of Network and Computer Applications* 91 (2017), pp. 1–13. DOI: https://doi.org/10.1016/j.jnca.2017.04.015 (cited on page 57).

[3] Machine Labs AI. *AI in Cybersecurity: A Comprehensive Overview of 2023*. Access: 08/11/2023. 2023. URL: https://machinelabs.ai/blog/ai-in-cybersecurity (cited on page 7).

[4] Cristina Alcaraz and Sherali Zeadally. 'Critical control system protection in the 21st century'. In: *Computer* 46.10 (2013), pp. 74–83 (cited on page 64).

[5] Cristina Alcaraz and Sherali Zeadally. 'Critical infrastructure protection: Requirements and challenges for the 21st century'. In: *International journal of critical infrastructure protection* 8 (2015), pp. 53–66 (cited on pages 63, 64).

[6] Monther Aldwairi, Musaab Hasan, and Zayed Balbahaith. 'Detection of drive-by download attacks using machine learning approach'. In: *Cognitive analytics: Concepts, methodologies, tools, and applications*. IGI Global, 2020, pp. 1598–1611 (cited on page 60).

[7] Rahaf Alkhadra et al. 'Solar Winds Hack: In-Depth Analysis and Countermeasures'. In: *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. 2021, pp. 1–7. DOI: 10.1109/ICCCNT51525.2021.9579611 (cited on pages 1, 33).

[8] Tiago A. Almeida and Akebo Yamakami. 'Content-based spam filtering'. In: *The 2010 International Joint Conference on Neural Networks (IJCNN)*. 2010, pp. 1–7. DOI: 10.1109/IJCNN.2010.5596569 (cited on page 150).

[9] Mohamed Alsharnouby, Furkan Alaca, and Sonia Chiasson. 'Why phishing still works: User strategies for combating phishing attacks'. In: *International Journal of Human-Computer Studies* 82 (2015), pp. 69–82 (cited on page 60).

[10] Laith Alzubaidi et al. 'Review of deep learning: concepts, CNN architectures, challenges, applications, future directions'. In: *Journal of Big Data* 8.1 (Mar. 2021). DOI: 10.1186/s40537-021-00444-8 (cited on page 47).

[11] Nisreen Ameen et al. 'Customer experiences in the age of artificial intelligence'. In: *Computers in Human Behavior* 114 (2021), p. 106548. DOI: https://doi.org/10.1016/j.chb.2020.106548 (cited on page 41).

[12] Uchenna P. Daniel Ani, Hongmei (Mary) He, and Ashutosh Tiwari. 'Review of cybersecurity issues in industrial critical infrastructure: manufacturing in perspective'. In: *Journal of Cyber Security Technology* 1.1 (Nov. 2016), pp. 32–74. DOI: 10.1080/23742917.2016.1252211 (cited on page 22).

[13] Mohd Ali Ansari and Dushyant Kumar Singh. 'Review of Deep Learning Techniques for Object Detection and Classification'. In: *Communication, Networks and Computing*. Springer Singapore, Oct. 2018, pp. 422–431. DOI: 10.1007/978-981-13-2372-0_37 (cited on page 47).

[14] Liat Antwarg et al. 'Explaining anomalies detected by autoencoders using Shapley Additive Explanations'. In: *Expert Systems with Applications* 186 (2021), p. 115736. DOI: https://doi.org/10.1016/j.eswa.2021.115736 (cited on page 88).

[15] Syed Muhammad Anwar et al. 'Medical Image Analysis using Convolutional Neural Networks: A Review'. In: *Journal of Medical Systems* 42.11 (Oct. 2018). DOI: 10.1007/s10916-018-1088-1 (cited on page 47).

[16] Giovanni Apruzzese, Luca Pajola, and Mauro Conti. 'The Cross-Evaluation of Machine Learning-Based Network Intrusion Detection Systems'. In: *IEEE Transactions on Network and Service Management* 19.4 (2022), pp. 5152–5169. DOI: 10.1109/TNSM.2022.3157344 (cited on page 55).

[17] Giovanni Apruzzese et al. 'On the effectiveness of machine and deep learning for cyber security'. In: *2018 10th International Conference on Cyber Conflict (CyCon)*. 2018, pp. 371–390. DOI: 10.23919/CYCON.2018.8405026 (cited on page 54).

[18] Giovanni Apruzzese et al. 'The Role of Machine Learning in Cybersecurity'. In: *Digital Threats* 4.1 (2023). DOI: 10.1145/3545574 (cited on page 54).

[19] Nalin Asanka Gamagedara Arachchilage, Steve Love, and Konstantin Beznosov. 'Phishing threat avoidance behaviour: An empirical investigation'. In: *Computers in Human Behavior* 60 (2016), pp. 185–197 (cited on page 61).

[20] W.C. Arnold and Gerald Tesauro. 'Automatically generated Win32 heuristic virus detection'. In: (Jan. 2000) (cited on page 119).

[21] ASEC. *ASEC Weekly Malware Statistics*. Access: 23/01/2024. 2023. URL: https://asec.ahnlab.com/en/52488/ (cited on page 118).

[22] Ömer Aslan Aslan and Refik Samet. 'A Comprehensive Review on Malware Detection Approaches'. In: *IEEE Access* 8 (2020), pp. 6249–6271. DOI: 10.1109/ACCESS.2019.2963724 (cited on page 54).

[23] Pinkesh Badjatiya et al. 'Deep learning for hate speech detection in tweets'. In: *Proceedings of the 26th international conference on World Wide Web companion*. 2017, pp. 759–760 (cited on page 59).

[24] Baezner, Marie and Robin, Patrice. *Stuxnet*. en. Tech. rep. 2017. DOI: 10.3929/ETHZ-B-000200661 (cited on page 28).

[25] Taimur Bakhshi and Bogdan Ghita. 'On Internet Traffic Classification: A Two-Phased Machine Learning Approach'. In: *Journal of Computer Networks and Communications* 2016 (2016), pp. 1–21. DOI: 10.1155/2016/2048302 (cited on page 109).

[26] Andrew G Barto and Thomas G Dietterich. 'Reinforcement learning and its relationship to supervised learning'. In: *Handbook of learning and approximate dynamic programming* 10 (2004), p. 9780470544785 (cited on page 45).

[27] Mohammad Mahdi Bejani and Mehdi Ghatee. 'A systematic review on overfitting control in shallow and deep neural networks'. In: *Artificial Intelligence Review* 54.8 (Mar. 2021), pp. 6391–6438. DOI: 10.1007/s10462-021-09975-1 (cited on page 124).

[28] Fabricio Benevenuto et al. 'Detecting spammers on twitter'. In: *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*. Vol. 6. 2010. 2010, p. 12 (cited on page 59).

[29] Ran Bi, Chengke Zhou, and Donald M Hepburn. 'Applying instantaneous SCADA data to artificial intelligence based power curve monitoring and WTG fault forecasting'. In: *2016 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)*. IEEE. 2016, pp. 176–181 (cited on page 64).

[30] Wenya Linda Bi et al. 'Artificial intelligence in cancer imaging: clinical challenges and applications'. In: *CA: a cancer journal for clinicians* 69.2 (2019), pp. 127–157 (cited on page 41).

[31] Marenglen Biba et al. 'Unsupervised Discretization Using Kernel Density Estimation.' In: *IJCAI*. Vol. 7. 2007, pp. 696–701 (cited on page 45).

[32] Bitsight. *How the Impact of WannaCry Ransomware Was Felt Around the World*. Access: 29/01/2024. 2017. URL: https://www.bitsight.com/blog/assessing-the-global-impact-of-wannacry-ransomware (cited on page 33).

[33] John M Blythe and Lynne Coventry. 'Cyber security games: a new line of risk'. In: *Entertainment Computing-ICEC 2012: 11th International Conference, ICEC 2012, Bremen, Germany, September 26-29, 2012. Proceedings 11*. Springer. 2012, pp. 600–603 (cited on page 61).

[34] Fatima Bouchama and Mostafa Kamal. 'Enhancing Cyber Threat Detection through Machine Learning-Based Behavioral Modeling of Network Traffic Patterns'. In: *International Journal of Business Intelligence and Big Data Analytics* 4.9 (Sept. 2021), pp. 1–9 (cited on page 67).

[35] Miles Brundage et al. *The Malicious Use of Artificial Intelligence: Forecasting, Prevention, and Mitigation*. 2018. DOI: 10.48550/ARXIV.1802.07228. URL: https://arxiv.org/abs/1802.07228 (cited on page 67).

[36] Bruce G Buchanan. 'A (very) brief history of artificial intelligence'. In: *Ai Magazine* 26.4 (2005), pp. 53–53 (cited on page 39).

[37] Adrián Campazas-Vega et al. 'Flow-Data Gathering Using NetFlow Sensors for Fitting Malicious-Traffic Detection Models'. In: *Sensors* 20.24 (2020), p. 7294. DOI: 10.3390/s20247294 (cited on pages 109, 110, 113).

[38] Guilherme O Campos et al. 'On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study'. In: *Data mining and knowledge discovery* 30 (2016), pp. 891–927 (cited on page 45).

[39] Capgemini. *Two in three organizations plan to deploy Artificial Intelligence to bolster their defense as soon as 2020*. Access: 29/01/2024. 2020. URL: https://www.capgemini.com/news/press-releases/ai-in-cybersecurity/ (cited on page 67).

[40] Valentín Carela-Español et al. 'Analysis of the impact of sampling on NetFlow traffic classification'. In: *Computer Networks* 55.5 (2011), pp. 1083–1099. DOI: 10.1016/j.comnet.2010.11.002 (cited on pages 109, 114).

[41] Rich Caruana and Alexandru Niculescu-Mizil. 'An empirical comparison of supervised learning algorithms'. In: *Proceedings of the 23rd international conference on Machine learning*. 2006, pp. 161–168 (cited on page 43).

[42] Pedro Casas, Johan Mazel, and Philippe Owezarski. 'Unsupervised Network Intrusion Detection Systems: Detecting the Unknown without Knowledge'. In: *Computer Communications* 35.7 (2012), pp. 772–783. DOI: 10.1016/j.comcom.2012.01.016 (cited on pages 109, 113).

[43] Davide Castelvecchi. 'Can we open the black box of AI?' In: *Nature* 538.7623 (Oct. 2016), pp. 20–23. DOI: 10.1038/538020a (cited on page 52).

[44] Patricia L Cavalcante et al. 'Centralized self-healing scheme for electrical distribution systems'. In: *IEEE transactions on smart grid* 7.1 (2015), pp. 145–155 (cited on page 65).

[45] Rajchada Chanajitt, Wantanee Viriyasitavat, and Kim-Kwang Raymond Choo. 'Forensic analysis and security assessment of Android m-banking apps'. In: *Australian Journal of Forensic Sciences* 50.1 (2018), pp. 3–19. DOI: 10.1080/00450618.2016.1182589 (cited on page 73).

[46] Junaid Ahsenali Chaudhry, Shafique Ahmad Chaudhry, and Robert G Rittenhouse. 'Phishing attacks and defenses'. In: *International journal of security and its applications* 10.1 (2016), pp. 247–256 (cited on page 59).

[47] Si-An Chen, Chun-Liang Li, and Hsuan-Tien Lin. 'A Unified View of cGANs with and without Classifiers'. In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 27566–27579 (cited on page 50).

[48] Jim X. Chen. 'The Evolution of Computing: AlphaGo'. In: *Computing in Science & Engineering* 18.4 (2016), pp. 4–7. DOI: 10.1109/MCSE.2016.74 (cited on page 40).

[49] Jing Chen et al. 'Uncovering the Face of Android Ransomware: Characterization and Real-Time Detection'. In: *IEEE Transactions on Inf. Forensics and Security* (2018). DOI: 10.1109/TIFS.2017.2787905 (cited on page 73).

[50] Qian Chen and Robert A. Bridges. 'Automated Behavioral Analysis of Malware: A Case Study of WannaCry Ransomware'. In: *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2017, pp. 454–460. DOI: 10.1109/ICMLA.2017.0-119 (cited on pages 1, 24).

[51] Tianying Chen, Jessica Hammer, and Laura Dabbish. 'Self-efficacy-based game design to encourage security behavior online'. In: *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 2019, pp. 1–6 (cited on page 61).

[52] K. R. Chowdhary. 'Natural Language Processing'. In: *Fundamentals of Artificial Intelligence*. Springer India, 2020, pp. 603–649. DOI: 10.1007/978-81-322-3972-7_19 (cited on page 48).

[53] Nabin Chowdhury and Vasileios Gkioulos. 'Cyber security training for critical infrastructure protection: A literature review'. In: *Computer Science Review* 40 (2021), p. 100361. DOI: https://doi.org/10.1016/j.cosrev.2021.100361 (cited on page 22).

[54] Hyunji Chung et al. 'Alexa, can I trust you?' In: *Computer* 50.9 (2017), pp. 100–104 (cited on page 61).

[55] CISA. *Principles and Approaches for Security-by-Design*. Access: 22/01/2024. 2023. URL: https://www.cisa.gov/news-events/news/next-chapter-secure-design (cited on page 106).

[56] Cisco. *Cisco Annual Internet Report (2018–2023) White Paper*. Access: 10/12/2023. 2020. URL: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html (cited on page 99).

[57] Jesse Clifton and Eric Laber. 'Q-learning: Theory and applications'. In: *Annual Review of Statistics and Its Application* 7 (2020), pp. 279–301 (cited on page 45).

[58] Cloudfare. *Sandboxing in Linux with zero lines of code*. Access: 24/01/2024. 2024. URL: https://blog.cloudflare.com/sandboxing-in-linux-with-zero-lines-of-code (cited on page 30).

[59] Melanie Coggan. 'Exploration and exploitation in reinforcement learning'. In: *Research supervised by Prof. Doina Precup, CRA-W DMP Project at McGill University* (2004) (cited on page 45).

[60] Nadia K Conroy, Victoria L Rubin, and Yimin Chen. 'Automatic deception detection: Methods for finding fake news'. In: *Proceedings of the association for information science and technology* 52.1 (2015), pp. 1–4 (cited on page 58).

[61] CoverLink. *Cyber Case Study: Sony Pictures Entertainment Hack*. 2021. URL: https://coverlink.com/case-study/sony-pictures-entertainment-hack/ (visited on 01/03/2024) (cited on pages 8, 33).

[62] Josh Cowls et al. 'The AI gambit: leveraging artificial intelligence to combat climate change—opportunities, challenges, and recommendations'. In: *Ai & Society* (2021), pp. 1–25 (cited on page 42).

[63] Darktrace. *Ransomware*. Access: 29/01/2024. 2020. URL: https://es.darktrace.com/products/email/use-cases/ransomware (cited on page 68).

[64] Lyubka Dencheva. 'Comparative analysis of Static application security testing (SAST) and Dynamic application security testing (DAST) by using open-source web application penetration testing tools'. PhD thesis. Dublin, National College of Ireland, 2022 (cited on page 30).

[65] Li Deng. 'Artificial intelligence in the rising wave of deep learning: The historical path and future outlook [perspectives]'. In: *IEEE Signal Processing Magazine* 35.1 (2018), pp. 180–177 (cited on page 40).

[66] Prateek Dewan and Ponnurangam Kumaraguru. 'Towards automatic real time identification of malicious posts on Facebook'. In: *2015 13th Annual Conference on Privacy, Security and Trust (PST)*. IEEE. 2015, pp. 85–92 (cited on page 58).

[67] Rahul Dey and Fathi M. Salem. 'Gate-variants of Gated Recurrent Unit (GRU) neural networks'. In: *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. 2017, pp. 1597–1600. DOI: 10.1109/MWSCAS.2017.8053243 (cited on page 48).

[68] Rachna Dhamija, J Doug Tygar, and Marti Hearst. 'Why phishing works'. In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 2006, pp. 581–590 (cited on page 60).

[69] GEN Digital. *Powering Digital Freedom*. Access: 24/01/2024. 2024. URL: https://www.gendigital.com/us/en/ (cited on page 24).

[70] Willian Dimitrov. 'The Impact of the Advanced Technologies over the Cyber Attacks Surface'. In: *Artificial Intelligence and Bioinspired Computational Methods*. Ed. by Radek Silhavy. Cham: Springer International Publishing, 2020, pp. 509–518 (cited on page 23).

[71] Herbert B Dixon Jr. 'Deepfakes: More frightening than photoshop on steroids'. In: *Judges J.* 58 (2019), p. 35 (cited on page 136).

[72] Rohan Doshi, Noah Apthorpe, and Nick Feamster. 'Machine Learning DDoS Detection for Consumer Internet of Things Devices'. In: *2018 IEEE Security and Privacy Workshops (SPW)*. 2018, pp. 29–35. DOI: 10.1109/SPW.2018.00013 (cited on page 55).

[73] Salim Dridi. 'Supervised learning-a systematic literature review'. In: (2021) (cited on page 43).

[74] DTEX. *The Global Leader for Insider Risk Management*. 2023. URL: https://www.dtexsystems.com/ (visited on 12/15/2023) (cited on page 13).

[75] Aeryn Dunmore et al. *Generative Adversarial Networks for Malware Detection: a Survey*. 2023. DOI: 10.48550/ARXIV.2302.08558. URL: https://arxiv.org/abs/2302.08558 (cited on page 50).

[76] Ahmed Elgammal et al. *CAN: Creative Adversarial Networks, Generating "Art" by Learning About Styles and Deviating from Style Norms*. 2017. DOI: 10.48550/ARXIV.1706.07068. URL: https://arxiv.org/abs/1706.07068 (cited on page 50).

[77] JHP Eloff et al. 'A Big Data Science Experiment–Identity Deception Detection'. In: *2015 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE. 2015, pp. 416–419 (cited on page 59).

[78] Deniz Engin, Anil Genc, and Hazim Kemal Ekenel. 'Cycle-Dehaze: Enhanced CycleGAN for Single Image Dehazing'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2018 (cited on page 50).

[79] Interesting Engineering. *What is deepfake technology and how does it work?* Access: 18/12/2023. 2022. URL: https://interestingengineering.com/culture/deepfake-technology-how-work (cited on page 50).

[80] Yigit Erkal, Mustafa Sezgin, and Sedef Gunduz. 'A new cyber security alert system for twitter'. In: *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE. 2015, pp. 766–770 (cited on page 59).

[81] Emre Erturk. 'A case study in open source software security and privacy: Android adware'. In: *World Congress on Internet Security (WorldCIS-2012)*. 2012 (cited on page 73).

[82] Secretaria de Estado de Seguridad - Ministerio del Interior España. *Informe sobre la cibercriminalidad en España 2022*. 2022. URL: https://estadisticasdecriminalidad.ses.mir.es/publico/portalestadistico/publicaciones.html (visited on 01/03/2024) (cited on pages 2, 3).

[83] Jianqing Fan et al. 'A theoretical analysis of deep Q-learning'. In: *Learning for dynamics and control*. PMLR. 2020, pp. 486–489 (cited on page 45).

[84] Steven Feldstein. 'How artificial intelligence systems could threaten democracy'. In: *The Conversation* (2019) (cited on page 136).

[85] Fortinet. *Fortinet 2023 Global Cyber Skills Gap Report Finds More Needs to be Done to Untap New Talent*. Access: 08/11/2023. 2023. URL: https://www.fortinet.com/blog/industry-trends/skills-gap-report-untap-talent (cited on page 6).

[86] World Economic Forum. *The cybersecurity skills gap is a real threat — here's how to address it*. Access: 08/11/2023. 2023. URL: https://www.weforum.org/agenda/2023/05/the-cybersecurity-skills-gap-is-a-real-threat-heres-how-to-address-it/ (cited on page 6).

[87] Jeremy Frank. 'Artiicial Intelligence and Intrusion Detection: Current and Future Directions'. In: 1994 (cited on page 53).

[88] Ohad Fried et al. 'Text-based editing of talking-head video'. In: *ACM Transactions on Graphics (TOG)* 38.4 (2019), pp. 1–14 (cited on page 138).

[89] Ganesh Gajula et al. 'A Machine Learning Based Adult Content Detection Using Support Vector Machine'. In: *2020 7th International Conference on Computing for Sustainable Global Development (INDIACom)*. 2020, pp. 181–185. DOI: 10.23919/INDIACom49435.2020.9083700 (cited on page 139).

[90] Andrew Garbett et al. 'Anti-social media: communicating risk through open data, crime maps and locative media'. In: *HCI* (2014), pp. 145–152 (cited on pages 61, 62).

[91] Christian Garbin, Xingquan Zhu, and Oge Marques. 'Dropout vs. batch normalization: an empirical study of their impact to deep learning'. In: *Multimedia Tools and Applications* 79.19–20 (Jan. 2020), pp. 12777–12815. DOI: 10.1007/s11042-019-08453-9 (cited on page 47).

[92] Holly Ann Garnett and Toby S. James. 'Cyber Elections in the Digital Age: Threats and Opportunities of Technology for Electoral Integrity'. In: *Election Law Journal: Rules, Politics, and Policy* 19.2 (June 2020), pp. 111–126. DOI: 10.1089/elj.2020.0633 (cited on page 24).

[93] GeekforGeeks. *How to Prevent Man In the Middle Attack?* Access: 17/01/2024. 2021. URL: https://www.geeksforgeeks.org/how-to-prevent-man-in-the-middle-attack/ (cited on page 97).

[94] Hossein Gholamalinezhad and Hossein Khosravi. 'Pooling Methods in Deep Neural Networks, a Review'. In: *ArXiv* abs/2009.07485 (2020) (cited on page 142).

[95] Daniel Gibert, Carles Mateu, and Jordi Planes. 'The rise of machine learning for detection and classification of malware: Research developments, trends and challenges'. In: *Journal of Network and Computer Applications* 153 (2020), p. 102526. DOI: https://doi.org/10.1016/j.jnca.2019.102526 (cited on page 54).

[96] Shlok Gilda. 'Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection'. In: *2017 IEEE 15th student conference on research and development (SCOReD)*. IEEE. 2017, pp. 110–115 (cited on pages 58, 59).

[97] Ben Goertzel and Cassio Pennachin. *Artificial general intelligence*. Vol. 2. Springer, 2007 (cited on page 38).

[98] Zorica Golić et al. 'Finance and artificial intelligence: The fifth industrial revolution and its impact on the financial sector'. In: *Zbornik radova Ekonomskog fakulteta u Istočnom Sarajevu* 19 (2019), pp. 67–81 (cited on page 41).

[99] José María Gómez Hidalgo et al. 'Content based SMS spam filtering'. In: *Proceedings of the 2006 ACM Symposium on Document Engineering*. DocEng '06. Amsterdam, The Netherlands: Association for Computing Machinery, 2006, pp. 107–114. DOI: 10.1145/1166160.1166191 (cited on page 150).

[100] Maoguo Gong et al. 'Evolving Deep Neural Networks via Cooperative Coevolution With Backpropagation'. In: *IEEE Transactions on Neural Networks and Learning Systems* 32.1 (2021), pp. 420–434. DOI: 10.1109/TNNLS.2020.2978857 (cited on page 124).

[101] Luis F Gonzalez et al. 'Unmanned aerial vehicles (UAVs) and artificial intelligence revolutionizing wildlife monitoring and conservation'. In: *Sensors* 16.1 (2016), p. 97 (cited on page 42).

[102] Ian Goodfellow et al. 'Generative adversarial networks'. In: *Commun. ACM* 63.11 (2020), pp. 139–144. DOI: 10.1145/3422622 (cited on page 49).

[103] Google. *AlphaGo*. Access: 29/01/2024. 2018. URL: https://deepmind.google/technologies/alphago/ (cited on page 40).

[104] Alex Graves and Jürgen Schmidhuber. 'Framewise phoneme classification with bidirectional LSTM and other neural network architectures'. In: *Neural networks* 18.5-6 (2005), pp. 602–610 (cited on page 152).

[105] Maanak Gupta et al. 'From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy'. In: *IEEE Access* 11 (2023), pp. 80218–80245. DOI: 10.1109/ACCESS.2023.3300381 (cited on page 67).

[106] Ilse van de Haar, Cecilie Pedersen Broberg, and Ifigenia Doshoris. 'How Artificial Intelligence is changing The Relationship Between The Consumer and Brand in The Music Industry'. In: *LBMG Strategic Brand Management-Masters Paper Series* (2019) (cited on page 41).

[107] HackerNoon. *The Dangers of DeepFake Technology*. Access: 29/12/2023. 2023. URL: https://hackernoon.com/the-dangers-of-deepfake-technology-exploring-the-potential-risks-of-ai-generated-videos-and-images (cited on page 135).

[108] Reda Mohamed Hamou, Abdelmalek Amine, and Amine Boudia. 'A New Meta-Heuristic Based on Social Bees for Detection and Filtering of Spam'. In: *International Journal of Applied Metaheuristic Computing* 4.3 (July 2013), pp. 15–33. DOI: 10.4018/ijamc.2013070102 (cited on page 150).

[109] Mahdi Hashemi. 'Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation'. In: *Journal of Big Data* 6.1 (Nov. 2019), p. 98. DOI: 10.1186/s40537-019-0263-7 (cited on page 80).

[110] Daojing He et al. 'A Comprehensive Detection Method for the Lateral Movement Stage of APT Attacks'. In: *IEEE Internet of Things Journal* (2023), pp. 1–1. DOI: 10.1109/JIOT.2023.3322412 (cited on page 5).

[111] Kaiming He et al. 'Deep Residual Learning for Image Recognition'. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016). DOI: 10.1109/cvpr.2016.90 (cited on page 112).

[112] Xin He, Kaiyong Zhao, and Xiaowen Chu. 'AutoML: A survey of the state-of-the-art'. In: *Knowledge-Based Systems* 212 (2021), p. 106622. DOI: https://doi.org/10.1016/j.knosys.2020.106622 (cited on page 50).

[113] Maurice Hendrix, Ali Al-Sherbaz, and Bloom Victoria. 'Game based cyber security training: are serious games suitable for cyber security training?' In: *International Journal of Serious Games* 3.1 (2016), pp. 53–61 (cited on page 61).

[114] Richard Hill. 'Dealing with cyber security threats: International cooperation, ITU, and WCIT'. In: *2015 7th International Conference on Cyber Conflict: Architectures in Cyberspace*. IEEE. 2015, pp. 119–134 (cited on page 37).

[115] Hanan Hindy et al. 'Improving SIEM for critical SCADA water infrastructures using machine learning'. In: *International Workshop on Security and Privacy Requirements Engineering*. Springer. 2018, pp. 3–19 (cited on page 64).

[116] Sepp Hochreiter and Jürgen Schmidhuber. 'Long short-term memory'. In: *Neural computation* 9.8 (1997), pp. 1735–1780 (cited on page 152).

[117] Rick Hofstede et al. 'Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX'. In: *IEEE Communications Surveys & Tutorials* 16.4 (2014), pp. 2037–2064. DOI: 10.1109/COMST.2014.2321898 (cited on page 108).

[118] Gao Huang et al. 'Densely connected convolutional networks'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4700–4708 (cited on page 142).

[119] Hanan Hussain, P. S. Tamizharasan, and C. S. Rahul. 'Design possibilities and challenges of DNN models: a review on the perspective of end devices'. In: *Artificial Intelligence Review* 55.7 (Jan. 2022), pp. 5109–5167. DOI: 10.1007/s10462-022-10138-z (cited on page 124).

[120] IBM. *Cost of a Data Breach Report 2020*. Access: 29/01/2024. 2020. URL: https://www.ibm.com/security/digital-assets/cost-data-breach-report/1Cost%20of%20a%20Data%20Breach%20Report%202020.pdf (cited on page 67).

[121] InfoBae. *La inteligencia artificial que crea contenido para adultos basado en personajes de anime*. Access: 29/12/2023. 2022. URL: https://www.infobae.com/america/tecno/2022/11/19/inteligencia-artificial-seria-usada-para-generar-contenido-para-adultos-basados-en-personajes-de-anime/ (cited on page 135).

[122] ISACA. *An Executive View of Key Cybersecurity Trends and Challenges in 2023*. Access: 08/11/2023. 2023. URL: https://www.isaca.org/resources/news-and-trends/industry-news/2023/an-executive-view-of-key-cybersecurity-trends-and-challenges-in-2023 (cited on page 6).

[123] Francisco Jáñez-Martino et al. 'A review of spam email detection: analysis of spammer strategies and the dataset shift problem'. In: *Artificial Intelligence Review* 56.2 (May 2022), pp. 1145–1173. DOI: 10.1007/s10462-022-10195-4 (cited on page 148).

[124] Amir Javed, Pete Burnap, and Omer Rana. 'Prediction of drive-by download attacks on twitter'. In: *Information Processing & Management* 56.3 (2019), pp. 1133–1145 (cited on page 60).

[125] Hongbo Jiang et al. 'Lightweight application classification for network management'. In: *Proceedings of the 2007 SIGCOMM workshop on Internet network management - INM 07* (2007). DOI: 10.1145/1321753.1321771 (cited on page 109).

[126] Rong Jiang et al. 'Efficient self-healing group key management with dynamic revocation and collusion resistance for SCADA in smart grid'. In: *Security and communication networks* 8.6 (2015), pp. 1026–1039 (cited on page 65).

[127] KA1D0. *Android Malware Analysis - DroidDream*. Access: 08/12/2023. 2019. URL: https://nikhilh20.medium.com/android-malware-analysis-droiddream-d06fc0d87bd2 (cited on page 75).

[128] Soner Can Kalkan et al. 'Image Enhancement Effects On Adult Content Classification'. In: *2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*. 2020, pp. 1–6. DOI: 10.1109/INISTA49547.2020.9194646 (cited on page 139).

[129] Yoshiki Kanda et al. 'ADMIRE: Anomaly detection method using entropy-based PCA with three-step sketches'. In: *Computer Communications* 36.5 (2013), pp. 575–588. DOI: 10.1016/j.comcom.2012.12.002 (cited on pages 109, 113).

[130] Shubhra Kanti Karmaker ("Santu") et al. 'AutoML to Date and Beyond: Challenges and Opportunities'. In: *ACM Comput. Surv.* 54.8 (2021). DOI: 10.1145/3470918 (cited on page 50).

[131] Andrej Karpathy, Justin Johnson, and Li Fei-Fei. *Visualizing and Understanding Recurrent Networks*. 2015. DOI: 10.48550/ARXIV.1506.02078. URL: https://arxiv.org/abs/1506.02078 (cited on page 152).

[132] Tero Karras et al. 'Analyzing and Improving the Image Quality of StyleGAN'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020 (cited on page 50).

[133] Vivek Kaul, Sarah Enslin, and Seth A Gross. 'History of artificial intelligence in medicine'. In: *Gastrointestinal endoscopy* 92.4 (2020), pp. 807–812 (cited on page 39).

[134] Arash Keshavarzi Arshadi et al. 'Artificial intelligence for COVID-19 drug discovery and vaccine development'. In: *Frontiers in Artificial Intelligence* (2020), p. 65 (cited on page 42).

[135] Houssain Kettani and Polly Wainwright. 'On the Top Threats to Cyber Systems'. In: *2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT)*. 2019, pp. 175–179. DOI: 10.1109/INFOCT.2019.8711324 (cited on page 31).

[136] Houssain Kettani and Polly Wainwright. 'On the Top Threats to Cyber Systems'. In: *2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT)*. 2019, pp. 175–179. DOI: 10.1109/INFOCT.2019.8711324 (cited on page 54).

[137] Mirza Golam Kibria et al. 'Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks'. In: *IEEE access* 6 (2018), pp. 32328–32338 (cited on pages 38, 41).

[138] Hyeongwoo Kim et al. 'Deep video portraits'. In: *ACM Transactions on Graphics (TOG)* 37.4 (2018), pp. 1–14 (cited on page 138).

[139] Nari Kim et al. 'A comparison between major artificial intelligence models for crop yield prediction: Case study of the midwestern United States, 2006–2015'. In: *ISPRS International Journal of Geo-Information* 8.5 (2019), p. 240 (cited on page 42).

[140] Ben Kirman, Conor Linehan, and Shaun Lawson. 'Reorienting geolocation data through mischievous design'. In: *Funology 2: From Usability to Enjoyment* (2018), pp. 225–240 (cited on page 61).

[141] ANTON KIVVA. *IT threat evolution in Q2 2023. Mobile statistics*. Access: 08/01/2024. 2023. URL: https://securelist.com/it-threat-evolution-q2-2023-mobile-statistics/110427/ (cited on page 73).

[142] Amit Klein. 'DOM based cross site scripting or XSS of the third kind'. In: *Web Application Security Consortium, Articles* 4 (2005), pp. 365–372 (cited on page 25).

[143] Jens Kober, J Andrew Bagnell, and Jan Peters. 'Reinforcement learning in robotics: A survey'. In: *The International Journal of Robotics Research* 32.11 (2013), pp. 1238–1274 (cited on page 41).

[144] Iryna Korshunova et al. 'Fast face-swap using convolutional neural networks'. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 3677–3685 (cited on page 136).

[145] Neeraj Kumar et al. 'An intelligent approach for building a secure decentralized public key infrastructure in VANET'. In: *Journal of Computer and System Sciences* 81.6 (2015), pp. 1042–1058 (cited on page 62).

[146] Terran Lane. 'An Application of Machine Learning to Anomaly Detection'. In: 1997 (cited on page 54).

[147] Stefan Larsson and Fredrik Heintz. 'Transparency in artificial intelligence'. In: *Internet Policy Review* 9.2 (May 2020). DOI: 10.14763/2020.2.1469 (cited on page 138).

[148] Chuanhuang Li et al. 'Detection and defense of DDoS attack–based on deep learning in OpenFlow-based SDN'. In: *International Journal of Communication Systems* 31.5 (2018), e3497 (cited on pages 56, 57).

[149] Jenny S Li et al. 'A comparison of classifiers and features for authorship authentication of social networking messages'. In: *Concurrency and Computation: Practice and Experience* 29.14 (2017), e3918 (cited on page 59).

[150] Lingzhi Li et al. 'Faceshifter: Towards high fidelity and occlusion aware face swapping'. In: *arXiv preprint arXiv:1912.13457* (2019) (cited on page 137).

[151] Ning Li, Martin Shepperd, and Yuchen Guo. 'A systematic review of unsupervised learning techniques for software defect prediction'. In: *Information and Software Technology* 122 (2020), p. 106287 (cited on page 44).

[152] Qing Li et al. 'Medical image classification with convolutional neural network'. In: *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. 2014, pp. 844–848. DOI: 10.1109/ICARCV.2014.7064414 (cited on page 142).

[153] Xufang Li, Peter K.K. Loh, and Freddy Tan. 'Mechanisms of Polymorphic and Metamorphic Viruses'. In: *2011 European Intelligence and Security Informatics Conference*. 2011, pp. 149–154. DOI: 10.1109/EISIC.2011.77 (cited on page 120).

[154] Yibin Li et al. 'Intelligent cryptography approach for secure distributed big data storage in cloud computing'. In: *Information Sciences* 387 (2017), pp. 103–115 (cited on page 62).

[155] Zewen Li et al. 'A survey of convolutional neural networks: analysis, applications, and prospects'. In: *IEEE transactions on neural networks and learning systems* (2021) (cited on page 46).

[156] Reyner Aranta Lika et al. 'NotPetya: Cyber Attack Prevention through Awareness via Gamification'. In: *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*. 2018, pp. 1–6. DOI: 10.1109/ICSCEE.2018.8538431 (cited on page 24).

[157] Peter Likarish, Eunjin Jung, and Insoon Jo. 'Obfuscated malicious javascript detection using classification techniques'. In: *2009 4th International Conference on Malicious and Unwanted Software (MALWARE)*. IEEE. 2009, pp. 47–54 (cited on page 61).

[158] Xiang Ling et al. 'Adversarial attacks against Windows PE malware detection: A survey of the state-of-the-art'. In: *Computers & Security* 128 (2023), p. 103134. DOI: https://doi.org/10.1016/j.cose.2023.103134 (cited on page 118).

[159] Kunlin Liu et al. 'Deepfacelab: Integrated, flexible and extensible face-swapping framework'. In: *Pattern Recognition* 141 (2023), p. 109628. DOI: https://doi.org/10.1016/j.patcog.2023.109628 (cited on page 137).

[160] Liu Liu et al. 'Detecting and Preventing Cyber Insider Threats: A Survey'. In: *IEEE Communications Surveys & Tutorials* 20.2 (2018), pp. 1397–1417. DOI: 10.1109/COMST.2018.2800740 (cited on page 105).

[161] Qiyu Liu. 'Comparisons of Conventional Computing and Quantum Computing Approaches'. In: *Highlights in Science, Engineering and Technology* 38 (Mar. 2023), pp. 502–507. DOI: 10.54097/hset.v38i.5875 (cited on page 35).

[162] Weibo Liu et al. 'A survey of deep neural network architectures and their applications'. In: *Neurocomputing* 234 (2017), pp. 11–26. DOI: 10.1016/j.neucom.2016.12.038 (cited on pages 125, 142).

[163] Xiang Liu, Ziyang Tang, and Baijian Yang. 'Predicting Network Attacks with CNN by Constructing Images from NetFlow Data'. In: *2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)* (2019). DOI: 10.1109/bigdatasecurity-hpsc-ids.2019.00022 (cited on pages 109–111, 113).

[164] Yang Liu and Yi-Fang Wu. 'Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks'. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1. 2018 (cited on page 58).

[165] Daniel Loebenberger and R Wielputz. 'Evolution! from creeper to storm'. In: *Presentation for the Seminar on" Malware*. 2006, pp. 1–7 (cited on page 23).

[166] Shao-An Lu. *faceswap-GAN*. Access: 3/01/2024. 2018. URL: https://github.com/shaoanlu/faceswap-GAN (cited on page 137).

[167] Yang Lu and Li Da Xu. 'Internet of Things (IoT) Cybersecurity Research: A Review of Current Research Topics'. In: *IEEE Internet of Things Journal* 6.2 (2019), pp. 2103–2115. DOI: 10.1109/JIOT.2018.2869847 (cited on page 36).

[168] Brady D Lund and Ting Wang. 'Chatting about ChatGPT: how may AI and GPT impact academia and libraries?' In: *Library Hi Tech News* 40.3 (2023), pp. 26–29 (cited on page 41).

[169] Gordon Lyon. *Nmap Tool*. Access: 24/01/2024. 2024. URL: https://nmap.org/docs.html (cited on page 27).

[170] Vijay M and Indumathi G. 'A highly secure Multi- Factor authentication system using biometrics to enhance privacy in Internet of Things (IOT)'. In: *International Research Journal of Multidisciplinary Technovation* (Nov. 2019), pp. 26–34. DOI: 10.34256/irjmtcon4 (cited on page 24).

[171] Kevin Macnish and Jeroen van der Ham. 'Ethics in cybersecurity research and practice'. In: *Technology in Society* 63 (2020), p. 101382. DOI: https://doi.org/10.1016/j.techsoc.2020.101382 (cited on page 13).

[172] Samaneh Mahdavifar et al. 'Dynamic Android Malware Category Classification using Semi-Supervised Deep Learning'. In: *2020 IEEE Intl Conf on Cyber Sci. and Techn. Con. (CyberSciTech)*. 2020. DOI: 10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00094 (cited on page 81).

[173] Polina Mamoshina et al. 'Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare'. In: *Oncotarget* 9.5 (2018), p. 5665 (cited on page 63).

[174] Pedro Marcelino. *Transfer learning from pre-trained models*. Access: 11/12/2023. 2018. URL: https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751 (cited on page 49).

[175] McAfee. *Worry-free, all-in-one online protection for your identity*. Access: 24/01/2024. 2024. URL: https://www.mcafee.com/ (cited on page 24).

[176] John McCarthy. 'Generality in artificial intelligence'. In: *Communications of the ACM* 30.12 (1987), pp. 1030–1035 (cited on page 39).

[177] Wes Mckinney. 'Pandas: a Foundational Python Library for Data Analysis and Statistics'. In: *Python High Performance Science Computer* (Jan. 2011) (cited on page 111).

[178] Misha Mehra and Dhawal Pandey. 'Event triggered malware: A new challenge to sandboxing'. In: *2015 Annual IEEE India Conference (INDICON)*. 2015, pp. 1–6. DOI: 10.1109/INDICON.2015.7443327 (cited on page 75).

[179] Yajie Miao, Mohammad Gowayyed, and Florian Metze. 'EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding'. In: *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. 2015, pp. 167–174. DOI: 10.1109/ASRU.2015.7404790 (cited on page 48).

[180] Agnieszka Mikołajczyk and Michał Grochowski. 'Data augmentation for improving deep learning in image classification problem'. In: *2018 international interdisciplinary PhD workshop (IIPhDW)*. IEEE. 2018, pp. 117–122 (cited on page 141).

[181] UCI ML. *SMS Spam Collection Dataset*. Access: 11/11/2023. 2017. URL: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset (cited on page 154).

[182] Mahmood Moghimi and Ali Yazdian Varjani. 'New rule-based phishing detection method'. In: *Expert systems with applications* 53 (2016), pp. 231–242 (cited on page 60).

[183] T Roosefert Mohan et al. 'Intelligent machine learning based total productive maintenance approach for achieving zero downtime in industrial machinery'. In: *Computers & Industrial Engineering* 157 (2021), p. 107267 (cited on page 42).

[184] Savita Mohurle and Manisha Patil. 'A brief study of wannacry threat: Ransomware attack 2017'. In: *International journal of advanced research in computer science* 8.5 (2017), pp. 1938–1940 (cited on page 32).

[185] Narcisa Roxana Moşteanu. 'Challenges for organizational structure and design as a result of digitalization and cybersecurity'. In: *The Business & Management Review* 11.1 (2020), pp. 278–286 (cited on page 36).

[186] Miranda Mowbray. 'The twittering machine.' In: *WEBIST (2)*. 2010, pp. 299–304 (cited on page 59).

[187] Nikesh Muthukrishnan et al. 'Brief history of artificial intelligence'. In: *Neuroimaging Clinics* 30.4 (2020), pp. 393–399 (cited on page 39).

[188] Abhijit Kumar Nag and Dipankar Dasgupta. 'An adaptive approach for continuous multi-factor authentication in an identity eco-system'. In: *Proceedings of the 9th Annual Cyber and Information Security Research Conference*. 2014, pp. 65–68 (cited on page 62).

[189] Satoshi Nakamoto. 'Bitcoin: A peer-to-peer electronic cash system'. In: (2008) (cited on page 63).

[190] Simone Natale and Andrea Ballatore. 'Imagining the thinking machine: Technological myths and the rise of artificial intelligence'. In: *Convergence* 26.1 (2020), pp. 3–18 (cited on page 40).

[191] MIT News. *System predicts 85 percent of cyber-attacks using input from human experts*. Access: 29/01/2024. 2016. URL: https://news.mit.edu/2016/ai-system-predicts-85-percent-cyber-attacks-using-input-human-experts-0418 (cited on page 67).

[192] NIST. *CVE 2019-0708*. Access: 24/01/2024. 2019. URL: https://nvd.nist.gov/vuln/detail/cve-2019-0708 (cited on page 28).

[193] Quamar Niyaz, Weiqing Sun, and Ahmad Y Javaid. 'A deep learning based DDoS detection system in software-defined networking (SDN)'. In: *arXiv preprint arXiv:1611.07400* (2016) (cited on page 56).

[194] Sven Nyholm and Jilles Smids. 'The ethics of accident-algorithms for self-driving cars: An applied trolley problem?' In: *Ethical theory and moral practice* 19.5 (2016), pp. 1275–1289 (cited on page 52).

[195] Iulian Ogrezeanu et al. 'Privacy-Preserving and Explainable AI in Industrial Applications'. In: *Applied Sciences* 12.13 (June 2022), p. 6395. DOI: 10.3390/app12136395 (cited on page 138).

[196] Kyle Olszewski et al. 'Realistic dynamic facial textures from a single image using gans'. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 5429–5438 (cited on page 136).

[197] Aleksandr Ometov et al. 'Multi-Factor Authentication: A Survey'. In: *Cryptography* 2.1 (Jan. 2018), p. 1. DOI: 10.3390/cryptography2010001 (cited on page 24).

[198] Onfido. *Deepfake Report*. Access: 17/01/2024. 2023. URL: https://onfido.com/category/report/ (cited on page 135).

[199] H. Orman. 'The Morris worm: a fifteen-year perspective'. In: *IEEE Security & Privacy* 1.5 (2003), pp. 35–43. DOI: 10.1109/MSECP.2003.1236233 (cited on page 23).

[200] Aissam Outchakoucht, ES-SAMAALI Hamza, and Jean Philippe Leroy. 'Dynamic access control policy based on blockchain and machine learning for the internet of things'. In: *International journal of advanced Computer Science and applications* 8.7 (2017) (cited on page 63).

[201] Yin Minn Pa Pa et al. 'An Attacker's Dream? Exploring the Capabilities of ChatGPT for Developing Malware'. In: *Proceedings of the 16th Cyber Security Experimentation and Test Workshop*. CSET '23. , Marina del Rey, CA, USA, Association for Computing Machinery, 2023, pp. 10–18. DOI: 10.1145/3607505.3607513 (cited on page 104).

[202] Razvan Pascanu, Tomas Mikolov, and Y. Bengio. 'On the difficulty of training Recurrent Neural Networks'. In: *30th International Conference on Machine Learning, ICML 2013* (Nov. 2012) (cited on page 47).

[203] Adam Paszke et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019 (cited on page 112).

[204] Luis Perez and Jason Wang. 'The effectiveness of data augmentation in image classification using deep learning'. In: *arXiv preprint arXiv:1712.04621* (2017) (cited on page 141).

[205] Jochen Peter and Patti M. Valkenburg. 'Adolescents' Exposure to Sexually Explicit Material on the Internet'. In: *Communication Research* 33.2 (2006), pp. 178–204. DOI: 10.1177/0093650205285369 (cited on page 138).

[206] Jianpeng Qi et al. 'An effective and efficient hierarchical K-means clustering algorithm'. In: *International Journal of Distributed Sensor Networks* 13.8 (2017), p. 1550147717728627 (cited on page 44).

[207] Tashfiq Rahman et al. 'Human Factors in Cybersecurity: A Scoping Review'. In: *The 12th International Conference on Advances in Information Technology*. IAIT2021. ACM, June 2021. DOI: 10.1145/3468784.3468789 (cited on page 36).

[208] Hamza Rahmani, Nabil Sahli, and Farouk Kamoun. 'Distributed denial-of-service attack detection scheme-based joint-entropy'. In: *Security and Communication Networks* 5.9 (2012), pp. 1049–1061 (cited on page 57).

[209] Sanjeev Rao, Anil Kumar Verma, and Tarunpreet Bhatia. 'A review on social spam detection: Challenges, open issues, and future directions'. In: *Expert Systems with Applications* 186 (2021), p. 115742. DOI: https://doi.org/10.1016/j.eswa.2021.115742 (cited on page 148).

[210] Rapid7. *Metasploit: The world's most used penetration testing framework*. Access: 24/01/2024. 2024. URL: https://www.metasploit.com/ (cited on page 31).

[211] Rahul Raveendranath et al. 'Android malware attacks and countermeasures: Current and future directions'. In: *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*. 2014, pp. 137–143. DOI: 10.1109/ICCICCT.2014.6992944 (cited on page 75).

[212] Parvin Razzaghi, Karim Abbasi, and Pegah Bayat. 'Learning spatial hierarchies of high-level features in deep neural network'. In: *Journal of Visual Communication and Image Representation* 70 (2020), p. 102817 (cited on page 46).

[213] Via Resource. *The Cyber Security Skills Gap In 2023*. Access: 08/11/2023. 2023. URL: https://viaresource.com/knowledge-hub/clients/the-cyber-security-skills-gap-in-2023 (cited on page 6).

[214] Markus Riek, Rainer Bohme, and Tyler Moore. 'Measuring the Influence of Perceived Cybercrime Risk on Online Service Avoidance'. In: *IEEE Transactions on Dependable and Secure Computing* 13.2 (2016), pp. 261–273. DOI: 10.1109/TDSC.2015.2410795 (cited on page 1).

[215] Ryan Riley, Xuxian Jiang, and Dongyan Xu. 'Multi-aspect profiling of kernel rootkit behavior'. In: EuroSys '09. Nuremberg, Germany: Association for Computing Machinery, 2009, pp. 47–60. DOI: 10.1145/1519065.1519072 (cited on page 75).

[216] Shaun Riordan. 'The Geopolitics of Cyberspace: a Diplomatic Perspective'. In: *Brill Research Perspectives in Diplomacy and Foreign Policy* 3.3 (2018), pp. 1–84. DOI: 10.1163/24056006-12340011 (cited on page 8).

[217] Victoria L Rubin et al. 'Fake news or truth? using satirical cues to detect potentially misleading news'. In: *Proceedings of the second workshop on computational approaches to deception detection*. 2016, pp. 7–17 (cited on page 59).

[218] Ondrej Rysavy et al. 'A formal authorization framework for networked SCADA systems'. In: *2012 IEEE 19th International Conference and Workshops on Engineering of Computer-Based Systems*. IEEE. 2012, pp. 298–302 (cited on page 64).

[219] Mustafa Hassan Saad, Ahmed Serageldin, and Goda Ismaeel Salama. 'Android spyware disease and medication'. In: *2015 Second International Conf. on Inf. Security and Cyber Forensics (InfoSec)*. 2015, pp. 118–125. DOI: 10.1109/InfoSec.2015.7435516 (cited on page 73).

[220] Kashif Saleem et al. 'An intelligent information security mechanism for the network layer of WSN: BIOSARP'. In: *Computational Intelligence in Security for Information Systems: 4th International Conference, CISIS 2011, Held at IWANN 2011, Torremolinos-Málaga, Spain, June 8-10, 2011. Proceedings*. Springer. 2011, pp. 118–126 (cited on page 62).

[221] Hojjat Salehinejad et al. *Recent Advances in Recurrent Neural Networks*. 2018. DOI: 10.48550/ARXIV.1801.01078. URL: https://arxiv.org/abs/1801.01078 (cited on page 47).

[222] Kadam Vikas Samarthrao and Vandana M. Rohokale. 'A hybrid meta-heuristic-based multi-objective feature selection with adaptive capsule network for automated email spam detection'. In: *International Journal of Intelligent Robotics and Applications* 6.3 (Jan. 2022), pp. 497–521. DOI: 10.1007/s41315-021-00217-9 (cited on page 150).

[223] Igor Santos et al. 'Adult Content Filtering through Compression-Based Text Classification'. In: *International Joint Conference CISIS'12-ICEUTE12-SOCO12 Special Sessions*. Ed. by Álvaro Herrero et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 281–288 (cited on page 139).

[224] Igor Santos et al. 'An Empirical Study on Word Sense Disambiguation for Adult Content Filtering'. In: *International Joint Conference SOCO'14-CISIS'14-ICEUTE'14*. Ed. by José Gaviria de la Puerta et al. Cham: Springer International Publishing, 2014, pp. 537–544 (cited on page 139).

[225] Igor Santos et al. 'Twitter Content-Based Spam Filtering'. In: *International Joint Conference SOCO'13-CISIS'13-ICEUTE'13*. Ed. by Álvaro Herrero et al. Cham: Springer International Publishing, 2014, pp. 449–458 (cited on page 150).

[226] Panda Security. *Deepfake scams increased by 3000% in 2023*. Access: 17/01/2024. 2023. URL: https://www.pandasecurity.com/it/mediacenter/truffe-deepfake-aumentate/ (cited on page 136).

[227] Hichem Sedjelmaci and Sidi Mohammed Senouci. 'Smart grid Security: A new approach to detect intruders in a smart grid Neighborhood Area Network'. In: *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*. 2016, pp. 6–11. DOI: 10.1109/WINCOM.2016.7777182 (cited on pages 55, 56).

[228] Frank Seide et al. 'Feature engineering in Context-Dependent Deep Neural Networks for conversational speech transcription'. In: *2011 IEEE Workshop on Automatic Speech Recognition & Understanding* (2011). DOI: 10.1109/asru.2011.6163899 (cited on page 125).

[229] ShymalaGowri Selvaganapathy, Sudha Sadasivam, and Vinayakumar Ravi. 'A Review on Android Malware: Attacks, Countermeasures and Challenges Ahead'. In: *Journal of Cyber Security and Mobility* (Mar. 2021). DOI: 10.13052/jcsm2245-1439.1017 (cited on page 76).

[230] Chengcheng Shao et al. 'The spread of fake news by social bots'. In: *arXiv preprint arXiv:1707.07592* 96 (2017), p. 104 (cited on page 58).

[231] Ashu Sharma and Sanjay Kumar Sahay. 'Evolution and Detection of Polymorphic and Metamorphic Malwares: A Survey'. In: *CoRR* abs/1406.7061 (2014) (cited on page 120).

[232] Kamran Shaukat et al. 'A Survey on Machine Learning Techniques for Cyber Security in the Last Decade'. In: *IEEE Access* 8 (2020), pp. 222310–222354. DOI: 10.1109/ACCESS.2020.3041951 (cited on page 54).

[233] Victor RL Shen, Chin-Shan Wei, and Tony Tong-Ying Juang. 'Javascript malware detection using a high-level fuzzy petri net'. In: *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*. Vol. 2. IEEE. 2018, pp. 511–514 (cited on page 60).

[234] Nathan Shone et al. 'A Deep Learning Approach to Network Intrusion Detection'. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 2.1 (2018), pp. 41–50. DOI: 10.1109/tetci.2017.2772792 (cited on pages 109, 113).

[235] Babins Shrestha, Manar Mohamed, and Nitesh Saxena. 'ZEMFA: Zero-Effort Multi-Factor Authentication based on Multi-Modal Gait Biometrics'. In: *2019 17th International Conference on Privacy, Security and Trust (PST)*. 2019, pp. 1–10. DOI: 10.1109/PST47121.2019.8949032 (cited on page 24).

[236] Eun-A Sim et al. 'GANs and DCGANs for generation of topology optimization validation curve through clustering analysis'. In: *Advances in Engineering Software* 152 (2021), p. 102957. DOI: https://doi.org/10.1016/j.advengsoft.2020.102957 (cited on page 50).

[237] Ankush Singla, Elisa Bertino, and Dinesh Verma. 'Overcoming the Lack of Labeled Data: Training Intrusion Detection Models Using Transfer Learning'. In: *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*. 2019, pp. 69–74. DOI: 10.1109/SMARTCOMP.2019.00031 (cited on page 48).

[238] Kamilya Smagulova and Alex Pappachen James. 'A survey on LSTM memristive neural network architectures and applications'. In: *The European Physical Journal Special Topics* 228.10 (Oct. 2019), pp. 2313–2324. DOI: 10.1140/epjst/e2019-900046-x (cited on page 151).

[239] Spamhaus. *Live Botnet Threats Worldwide*. Access: 17/01/2024. 2024. URL: https://www.spamhaus.com/threat-map/ (cited on page 76).

[240] Martijn Spitters et al. 'Threat detection in tweets with trigger patterns and contextual cues'. In: *2014 IEEE Joint Intelligence and Security Informatics Conference*. IEEE. 2014, pp. 216–219 (cited on pages 58, 59).

[241] SpyCloud. *The Rise of Mobile Malware*. Access: 11/12/2023. 2023. URL: https://spycloud.com/blog/rise-of-mobile-malware/ (cited on page 74).

[242] Dan-Cristian Stanciu and Bogdan Ionescu. 'Autoencoder-based Data Augmentation for Deepfake Detection'. In: *Proceedings of the 2nd ACM International Workshop on Multimedia AI against Disinformation*. MAD '23. Thessaloniki, Greece: Association for Computing Machinery, 2023, pp. 19–27. DOI: 10.1145/3592572.3592840 (cited on page 137).

[243] StationX. *DDoS Attack Patterns Statistics*. Access: 11/12/2023. 2023. URL: https://www.stationx.net/ddos-statistics/ (cited on page 100).

[244] Statista. *Global mobile OS market share 2023*. 2023. URL: https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/ (visited on 02/15/2023) (cited on page 73).

[245] Richard S Sutton et al. 'Policy gradient methods for reinforcement learning with function approximation'. In: *Advances in neural information processing systems* 12 (1999) (cited on page 45).

[246] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. 'Synthesizing obama: learning lip sync from audio'. In: *ACM Transactions on Graphics (ToG)* 36.4 (2017), pp. 1–13 (cited on page 138).

[247] Mirac Suzgun, Yonatan Belinkov, and Stuart M. Shieber. 'On Evaluating the Generalization of LSTM Models in Formal Languages'. In: *CoRR* abs/1811.01001 (2018) (cited on page 151).

[248] Fabio Henrique Kiyoiti dos Santos Tanaka and Claus Aranha. *Data Augmentation Using GANs*. 2019. DOI: 10.48550/ARXIV.1904.09135. URL: https://arxiv.org/abs/1904.09135 (cited on page 50).

[249] Tuan A Tang et al. 'Deep learning approach for network intrusion detection in software defined networking'. In: *2016 international conference on wireless networks and mobile communications (WINCOM)*. IEEE. 2016, pp. 258–263 (cited on page 56).

[250] TechNative. *The Evolution of Mobile Malware*. Access: 11/12/2023. 2023. URL: https://technative.io/the-evolution-of-mobile-malware/ (cited on page 75).

[251] Tenable. *Nessus Product*. Access: 24/01/2024. 2024. URL: https://es-la.tenable.com/products/nessus (cited on page 27).

[252] Hiral Thadeshwar et al. 'Artificial Intelligence based Self-Driving Car'. In: *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP)*. 2020, pp. 1–5. DOI: 10.1109/ICCCSP49186.2020.9315223 (cited on pages 38, 41).

[253] Justus Thies et al. 'Face2face: Real-time face capture and reenactment of rgb videos'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2387–2395 (cited on page 137).

[254] Correo Total. *Qué hay al otro lado de su bandeja de entrada*. Access: 23/01/2024. 2023. URL: https://www.correototal.com/internet/estadisticas/que-hay-al-otro-lado-de-su-bandeja-de-entrada-20-estadisticas-de-spam-para-2023/ (cited on page 150).

[255] Enforcement Tracker. *GDPR Fines Stats*. Access: 26/01/2024. 2024. URL: https://www.enforcementtracker.com/?insights (cited on page 32).

[256] Quang Anh Tran, Frank Jiang, and Quang Minh Ha. 'Evolving Block-Based Neural Network and Field Programmable Gate Arrays for Host-Based Intrusion Detection System'. In: *2012 Fourth International Conference on Knowledge and Systems Engineering* (2012). DOI: 10.1109/kse.2012.31 (cited on pages 109, 113).

[257] Quang Anh Tran, Frank Jiang, and Jiankun Hu. 'A Real-Time NetFlow-based Intrusion Detection System with Improved BBNN and High-Frequency Field Programmable Gate Arrays'. In: *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications* (2012). DOI: 10.1109/trustcom.2012.51 (cited on pages 109, 113).

[258] Ashutosh Tripathi. *What Is The Main Difference Between Rnn And Lstm*. Access: 11/12/2023. 2021. URL: https://ashutoshtripathi.com/2021/07/02/what-is-the-main-difference-between-rnn-and-lstm-nlp-rnn-vs-lstm/ (cited on page 152).

[259] Michail Tsikerdekis and Sherali Zeadally. 'Online deception in social media'. In: *Communications of the ACM* 57.9 (2014), pp. 72–80 (cited on page 58).

[260] Stanford University. *Computer vision algorithm that can describe photos*. Access: 23/01/2024. 2014. URL: https://engineering.stanford.edu/magazine/article/stanford-team-creates-computer-vision-algorithm-can-describe-photos (cited on page 139).

[261] Cybersecurity Ventures. *Cybersecurity Jobs Report: 3.5 Million Unfilled Positions In 2025*. Access: 03/10/2023. 2023. URL: https://cybersecurityventures.com/jobs/ (cited on page 1).

[262] VirusShare. *VirusShare - Because Sharing is Caring*. Access: 10/12/2023. 2023. URL: https://virusshare.com/about (cited on page 127).

[263] Daniel Vlasic et al. 'Face transfer with multilinear models'. In: *ACM SIGGRAPH 2006 Courses*. 2006, 24–es (cited on page 137).

[264] Cheng Wang et al. 'Malware Detection Based on Suspicious Behavior Identification'. In: *2009 First International Workshop on Education Technology and Computer Science*. Vol. 2. 2009, pp. 198–202. DOI: 10.1109/ETCS.2009.306 (cited on page 119).

[265] Junjie Wang et al. 'Jsdc: A hybrid approach for javascript malware detection and classification'. In: *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*. 2015, pp. 109–120 (cited on page 61).

[266] Ting-Chun Wang et al. 'High-Resolution Image Synthesis and Semantic Manipulation With Conditional GANs'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018 (cited on page 50).

[267] Yequan Wang et al. 'Attention-based LSTM for aspect-level sentiment classification'. In: *Proceedings of the 2016 conference on empirical methods in natural language processing*. 2016, pp. 606–615 (cited on page 152).

[268] Zhiqiang Wang, Qian Liu, and Yaping Chi. 'Review of Android Malware Detection Based on Deep Learning'. In: *IEEE Access* 8 (2020), pp. 181102–181126. DOI: 10.1109/ACCESS.2020.3028370 (cited on page 77).

[269] Gaute Wangen. 'The Role of Malware in Reported Cyber Espionage: A Review of the Impact and Mechanism'. In: *Information* 6.2 (2015), pp. 183–211. DOI: 10.3390/info6020183 (cited on page 102).

[270] Information Week. *12 Ways to Approach the Cybersecurity Skills Gap Challenge in 2023*. Access: 08/11/2023. 2023. URL: https://www.informationweek.com/it-leadership/12-ways-to-approach-the-cybersecurity-skills-gap-challenge-in-2023 (cited on page 6).

[271] Jônatas Wehrmann et al. 'Adult content detection in videos with convolutional and recurrent neural networks'. In: *Neurocomputing* 272 (2018), pp. 432–438. DOI: https://doi.org/10.1016/j.neucom.2017.07.012 (cited on page 139).

[272] Jônatas Wehrmann et al. 'Adult content detection in videos with convolutional and recurrent neural networks'. In: *Neurocomputing* 272 (2018), pp. 432–438 (cited on page 139).

[273] Dave Wichers. *Unraveling some of the Mysteries around DOM Based XSS*. Access: 24/01/2024. 2012. URL: https://owasp.org/www-pdf-archive/Unraveling_some_Mysteries_around_DOM-based_XSS.pdf (cited on page 26).

[274] Marco A Wiering and Martijn Van Otterlo. 'Reinforcement learning'. In: *Adaptation, learning, and optimization* 12.3 (2012), p. 729 (cited on page 45).

[275] Philipp Winter, Eckehard Hermann, and Markus Zeilinger. 'Inductive Intrusion Detection in Flow-Based Network Data Using One-Class Support Vector Machines'. In: *2011 4th IFIP International Conference on New Technologies, Mobility and Security* (2011). DOI: 10.1109/ntms.2011.5720582 (cited on pages 109, 113).

[276] Computer World. *Unsung innovators: Gary Thuerk, the father of spam*. Access: 20/01/2024. 2007. URL: https://www.computerworld.com/article/2539767/unsung-innovators--gary-thuerk--the-father-of-spam.html (cited on page 147).

[277] Secure World. *Global Cybersecurity Skills Gap Still Widening Despite Growing Workforce*. Access: 08/11/2023. 2023. URL: https://www.secureworld.io/industry-news/global-cybersecurity-skills-gap-widening (cited on page 6).

[278] Station X. *Phishing Statistics*. Access: 23/01/2024. 2023. URL: https://www.stationx.net/phishing-statistics/ (cited on page 147).

[279] Tian Xia. 'A Constant Time Complexity Spam Detection Algorithm for Boosting Throughput on Rule-Based Filtering Systems'. In: *IEEE Access* 8 (2020), pp. 82653–82661. DOI: 10.1109/ACCESS.2020.2991328 (cited on page 149).

[280] Wei Xiong and Li Xiong. 'Smart contract based data trading mode using blockchain and machine learning'. In: *IEEE Access* 7 (2019), pp. 102331–102344 (cited on page 63).

[281] Runxin Xu et al. *Raise a Child in Large Language Model: Towards Effective and Generalizable Fine-tuning*. 2021. DOI: 10.48550/ARXIV.2109.05687. URL: https://arxiv.org/abs/2109.05687 (cited on page 49).

[282] Qiao Yan et al. 'Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges'. In: *IEEE communications surveys & tutorials* 18.1 (2015), pp. 602–622 (cited on page 56).

[283] Ping Yi et al. 'Puppet attack: A denial of service attack in advanced metering infrastructure network'. In: *Journal of Network and Computer Applications* 59 (2016), pp. 325–332. DOI: https://doi.org/10.1016/j.jnca.2015.04.015 (cited on page 55).

[284] Zibo Yi et al. 'Improving JavaScript Malware Classifier's Security against Evasion by Particle Swarm Optimization'. In: *2016 IEEE Trustcom/BigDataSE/ISPA*. IEEE. 2016, pp. 1734–1740 (cited on page 60).

[285] Yong Yu et al. 'A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures'. In: *Neural Computation* 31.7 (July 2019), pp. 1235–1270. DOI: 10.1162/neco_a_01199 (cited on page 151).

[286] Zhenlong Yuan, Yongqiang Lu, and Yibo Xue. 'DroidDetector: Android Malware Characterization and Detection Using Deep Learning'. In: *Tsinghua Science and Technology* 21.1 (2016), pp. 114–123. DOI: 10.1109/TST.2016.7399288 (cited on page 76).

[287] Muhammad Rehman Zafar and Naimul Khan. 'Deterministic Local Interpretable Model-Agnostic Explanations for Stable Explainability'. In: *Machine Learning and Knowledge Extraction* 3.3 (2021), pp. 525–541. DOI: 10.3390/make3030027 (cited on page 88).

[288] Florian Zandt. *Big Tech Invests Big in Cybersecurity*. 2022. URL: https://www.statista.com/chart/27088/gafam-spending-on-cybersecurity-deals-and-funding-per-year/ (visited on 01/05/2024) (cited on page 9).

[289] Florian Zandt. *The Most Prevalent Forms of Cyber Crime*. 2023. URL: https://www.statista.com/chart/30870/share-of-worldwide-cyber-attacks-by-type/ (visited on 01/04/2024) (cited on page 4).

[290] Xichen Zhang and Ali A. Ghorbani. 'Human Factors in Cybersecurity: Issues and Challenges in Big Data'. In: *Research Anthology on Privatizing and Securing Data*. IGI Global, 2021, pp. 1695–1725. DOI: 10.4018/978-1-7998-8954-0.ch082 (cited on page 36).

[291] Zhihe Zhao et al. 'EdgeML: An AutoML Framework for Real-Time Deep Learning on the Edge'. In: *Proceedings of the International Conference on Internet-of-Things Design and Implementation*. IoTDI '21. Charlottesvle, VA, USA: Association for Computing Machinery, 2021, pp. 133–144. DOI: 10.1145/3450268.3453520 (cited on page 51).

[292] Wang Zhenqi and Wang Xinyu. 'NetFlow Based Intrusion Detection System'. In: *2008 International Conference on MultiMedia and Information Technology* (2008). DOI: 10.1109/mmit.2008.213 (cited on page 109).

[293] Lu Zhou et al. 'Automatic fine-grained access control in SCADA by machine learning'. In: *Future Generation Computer Systems* 93 (2019), pp. 548–559 (cited on page 64).

[294] XueFei Zhou. 'Understanding the convolutional neural networks with gradient descent and backpropagation'. In: *Journal of Physics: Conference Series*. Vol. 1004. IOP Publishing. 2018, p. 012028 (cited on page 44).

[295] Frederik Zuiderveen Borgesius et al. 'Discrimination, artificial intelligence, and algorithmic decision-making'. In: *línea], Council of Europe* (2018) (cited on page 51).

[296] Zvelo. *Phishing Detection in Depth*. 2023. URL: https://zvelo.com/phishing-detection-in-depth/ (visited on 01/04/2024) (cited on page 4).