ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for Electronic Crime Investigation

(IJECI)

VOLUME: 7
ISSUE: 3 Jul-Sep 2023

Email ID: ijeci@lgu.edu.pk

Digital Forensics Research and Service Center
Lahore Garrison University, Lahore, Pakistan.

International Journal for Electronic Crime Investigation
Volume 7(3) Jul-Sep 2023

SCOPE OF THE JOURNAL

The International Journal for Crime Investigation IJECI is an innovative forum for researchers, scientists and engineers in all the domains of computer science, white Collar Crimes, Digital Forensics, Nano Forensics, Toxicology and related technology, Criminology, Criminal Justice and Criminal Behaviour Analysis. Moreover, the scope of the journal includes algorithm, high performance, Criminal Data Communication and Networks, pattern recognition, image processing, artificial intelligence, VHDL along with emerging domains like quantum computing, IoT, Hacking. The journal aims to provide an academic medium for emerging research trends in the general domain of crime investigation.

SUBMISSION OF ARTICLES

We invite articles with high quality research for publication in all areas of engineering, science and technology. All the manuscripts submitted for publication are first peer reviewed to make sure they are original, relevant and readable. Manuscripts should be submitted via email only.

To submit manuscripts by email with attach file is strongly encouraged, provided that the text, tables, and figures are included in a single Microsoft Word/Pdf file.

Contact: For all inquiries, regarding call for papers, submission of research articles and correspondence,kindly contact at this address:
IJECI, Sector C, DHA Phase-VI Lahore, Pakistan
Phone: +92- 042-37181823
Email: ijeci@lgu.edu.pk

International Journal for Electronic Crime Investigation

Volume 7(3) Jul-Sep 2023

CONTENTS

International Journal for Electronic Crime Investigation
Volume 7(3) Jul-Sep 2023

Editorial

Leveraging Technology to Combat White Collar Crime in Pakistan

Kaukab Jamal Zuberi

White collar crime, a term coined by sociologist Edwin Sutherland in 1939, refers to non-violent, financially motivated offenses typically committed by individuals, businesses, or government officials in positions of trust and authority. This form of crime encompasses a wide range of illicit activities, including fraud, embezzlement, corruption, money laundering, tax evasion, and insider trading, among others. While it may not be as sensationalized as street crime, white collar crime poses a significant threat to the economic and social fabric of a nation.

Pakistan, like many countries around the world, faces a considerable challenge in dealing with white collar crime. These offenses erode public trust, distort market dynamics, siphon off public funds, and hinder economic growth. However, combating white collar crime has proven to be a complex task, with traditional investigative methods often falling short. To address this challenge, Pakistan must embrace and harness the power of technology for more effective detection, investigation, and prevention of white collar crime.

The Need for Technology in Investigating White Collar Crime

White collar criminals have become increasingly sophisticated, exploiting technology to conceal their illicit activities. To combat these evolving threats, law enforcement agencies, regulatory bodies, and financial institutions in Pakistan must adapt and adopt cutting-edge technology. There are several compelling reasons why technology is indispensable in the fight against white collar crime.

1 Data Analytics and Pattern Recognition

One of the hallmarks of white collar crime is the manipulation of financial data. Technology can help investigators analyze large datasets quickly, identify irregularities, and detect suspicious patterns that may otherwise go unnoticed. Advanced analytics tools can trace the flow of money and connections between individuals and organizations, providing insights into money laundering, fraudulent schemes, and corruption networks.

2 Improved Surveillance

Modern surveillance technology, including closed-circuit television (CCTV) systems, facial recognition software, and data analytics, enables authorities to monitor high-risk areas and individuals involved in suspicious activities. This not only helps in tracking potential white collar criminals but also serves as a deterrent to such activities.

3 Digital Forensics

The digital age has given rise to a new breed of white collar criminals who leave digital footprints. Digital forensics allows investiga-

tors to recover and analyze electronic evidence from computers, smartphones, and other devices. This is essential in cases involving cybercrime, intellectual property theft, and embezzlement.

4 Cybersecurity

As the world becomes increasingly interconnected, the threat of cybercrime grows. Protecting sensitive financial and personal data is paramount. Technology can be employed to fortify cybersecurity measures, detect cyberattacks, and prevent data breaches.

5 Blockchain and Cryptocurrencies

White collar criminals often use cryptocurrencies for money laundering and illegal transactions. Understanding and utilizing blockchain technology can help track the movement of cryptocurrencies and identify those involved in financial crimes.

6 Transparency and Accountability

Technology can be instrumental in ensuring transparency and accountability in government, business, and financial institutions. Implementation of e-governance and electronic records can reduce the risk of corruption and embezzlement, making it harder for white collar criminals to operate with impunity.

7 International Cooperation

White collar crime often transcends national borders. Leveraging technology for cross-border data sharing, collaboration, and intelligence exchange is essential to effectively combat transnational financial crimes.

Challenges in Implementing Technology in White Collar Crime Investigations

While the benefits of integrating technology into white collar crime investigations are clear, there are challenges to its effective implementation in Pakistan:

1 Infrastructure and Funding

To deploy technology effectively, Pakistan must invest in the necessary infrastructure, equipment, and training. Financial constraints and resource limitations can hinder these efforts.

2 Skill Gaps

The success of technology-driven investigations depends on the availability of skilled personnel who can operate and interpret the technology. Training and developing a cadre of experts is a critical requirement.

3 Privacy Concerns

The use of surveillance technology, data analytics, and digital forensics must be carefully balanced with individual privacy rights and data protection laws. Striking the right balance can be challenging.

4 Cybersecurity Risks

As technology is employed to fight white collar crime, it is equally important to secure these technologies against hacking and cyber threats. A data breach or compromise of investigative tools could have dire consequences.

5 Legal Frameworks

Pakistan needs updated laws and regulations

that address the use of technology in investigations. Ensuring that these laws are both effective and protect individuals' rights is a delicate balance. The evidence law has the potential to be modified.

6 Institutional Resistance

Change is often met with resistance within organizations. The adoption of technology may face pushback from individuals and institutions reluctant to adapt to new methodologies.

Best Practices and Success Stories

Several countries around the world have successfully harnessed technology to combat white collar crime. Learning from their best practices and success stories can provide valuable insights for Pakistan:

1. United States - The Financial Crimes Enforcement Network (FinCEN) is a prime example of a government agency effectively using technology to combat money laundering and financial crimes. They use advanced analytics and data sharing to detect suspicious financial transactions.

2. United Kingdom - The UK's National Crime Agency (NCA) has implemented a sophisticated digital forensics program that helps in investigating financial crimes. They have also embraced cybersecurity initiatives to protect against cyber threats.

3. Singapore - Singapore has invested heavily in technology to monitor financial transactions and detect money laundering. Their collaboration with the private sector and use of data analytics has been instrumental in identifying and prosecuting white collar criminals.

4. Australia - The Australian Securities and Investments Commission (ASIC) uses technology for market surveillance and data analytics to detect insider trading and securities fraud. They have also established a national database to track beneficial ownership information.

5. India - The introduction of Aadhaar, a biometric identification system, has been a game-changer in India. It has made it significantly more difficult for individuals to engage in corruption and fraud.

Recommendations for Pakistan

To successfully leverage technology in the fight against white collar crime, Pakistan should consider the following recommendations:

1 Investment in Technology Infrastructure

Allocate adequate resources to establish a robust technology infrastructure that can support data analytics, digital forensics, cybersecurity, and surveillance systems.

2 Training and Capacity Building

Invest in training and capacity building programs to ensure law enforcement, regulatory bodies, and financial institutions have the skills to effectively use technology in investigations.

3 Legislative Reforms

Review and update existing laws and regulations to accommodate the use of technology in investigations while safeguarding individual

rights and data privacy.

4 Public-Private Partnerships

Foster collaboration between the government and the private sector to share data, intelligence, and best practices in combating white collar crime.

5 International Cooperation

Strengthen cooperation with international agencies and organizations to share intelligence and fight transnational financial crimes effectively.

6 Whistleblower Protection

Implement whistleblower protection laws and mechanisms to encourage individuals with inside knowledge of financial crimes to come forward.

7 Transparency Initiatives

Promote transparency and accountability through the use of e-governance, electronic records, and open data initiatives.

8 Cybersecurity Measures

Prioritize and invest in robust cybersecurity measures to protect investigative tools, data, and sensitive information from cyber threats.

Conclusion

The fight against white collar crime in Pakistan requires a paradigm shift in the way authorities approach investigations. Traditional methods are no longer sufficient to combat the evolving tactics of white-collar criminals. The integration of technology, including data analytics, surveillance, digital forensics, and cybersecurity, is imperative to level the playing field and protect the country's financial and social interests. It is recommended that all the stake holders ie Investigating Agencies, Prosecutors and Judiciary should be trained to use and understand the emerging technologies and their effectiveness in solving white collar crimes.

While there are challenges to implementing technology in white collar crime investigations, the benefits far outweigh the risks. By learning from successful international models and taking a proactive approach, Pakistan can establish itself as a leader in the fight against financial crimes. The goal is to foster an environment where white collar criminals are deterred by the certainty of apprehension and the severity of penalties, thus safeguarding the nation's economic and social well-being.

Aftab *et al. (IJECI)* 2023

(IJECI)
ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for
Electronic Crime Investigation

DOI: https://doi.org/10.54692/ijeci.2023.0703156

Research Article

Vol. 7 issue 3 Jul-Sep 2023

Modern Electronic and other Technologies to Combat New Wave of Terrorism and Criminal Activities

Prof Dr. Aftab Ahmad Malik[1], Dr. Waqar Azeem[2] and Dr. Mujtaba Asad[3]

[1]Ph.D (University of Kent, Canterbury England) ; M.Phil; MSc; LL.B. Professor Department of Criminology and Forensic Sciences, Lahore Garrison University, Lahore

[2]PhD; M.Phil, Head of Department Software Engineering, Lahore Garrison University, Lahore Formerly South Eastern Regional College, Down Patrick Ireland, UK.

[3]PhD (China), MPhil, MS, Senior Research Fellow, Department of Automation and Control Shanghai Jiao Tong University, Shanghai, China

Corresponding author: dr_aftab-malik@lgu.edu.pk

Received: June 07, 2023; **Accepted:** August 10,2023; **Published:** September 20, 2023

Abstract

Terrorists are using most recent technology to launch their attacks, therefore it is important for the suffering states like Pakistan to excessively practice the recent Electronic and other devices and technologies to safeguard and combat terrorism. By boosting security, intelligence, and surveillance capacities, electronic technology is essential in the fight against terrorism. The following electronic tools and systems are frequently employed to combat terrorism. In this paper we will focus on Artificial Intelligence, Biometric, Border Security Technology, Cybersecurity, Surveillance Technology, Communication and Encryption, use of Drones, Geospatial Technology in the context of terrorism. Apart from this we discuss about the new wave of terrorist attacks from foreign nationals and refugees living in Pakistan and their modus operandi and how to combat with them using Electronic Technology. More emphasis is required to be given to the Border Security Technology. It's crucial to remember that even if these electronic technologies are useful tools in counterterrorism efforts, they must be utilized legally and with due regard for people's civil liberties, privacy, and individual rights. A multidisciplinary strategy, including collaboration between numerous agencies, intelligence sharing, international cooperation, and complete policies and plans, is frequently necessary for counterterrorism efforts to be successful. Pakistan is facing three major problem of smuggling of commodities, foreign currency, and terrorism for across boarder.

Keywords: Terrorism, Criminal Data Bases of terrorists, Cyber Security and counter Terrorism.

1. Introduction

The use of science and technology plays a crucial role; [1] asserts that there are technological answers for practically all issues relating to terrorism. As technology has developed in recent years, criminals have also gained access to its tools and technical

know-how, particularly terrorists and gangsters engaged in white-collar crimes, bank robberies, robberies, and having information with which to hurt others. The terms revolutionary, sub-revolutionary, and establishment refer to the formation, creation, and launch of the three types of terrorism. There is a lot of "ideologically motivated terrorism" that happens. The use of the Internet, sophisticated intelligence gathering and analysis tools, and intelligent identification systems are just a few examples of how technology may be a vital aid in the fight against terrorism. Individual terrorist attacks are taking place all over the world, such as the shooting in a mosque in New Zealand and the recent incident in Texas, USA, when two families suffered significant losses. In May 2023, a second deadly shooting attack involving an 18-year-old kid took place in Farmington, New Mexico (in the northwest region of the state). In order to identify and discover terrorist groups and individuals working in public places, it is essential and crucial to use the most cutting-edge and inventive surveillance technologies combined with CCTV cameras. To prevent stress and suffering, the devices must be installed in parks, schools, stores, and other significant structures.

Some important policy features of US Department of State are relevant regarding transparency, Anti-corruption, Arms Control, Combating drugs and Crime and countering the terrorism. Cyber issues. Programs of public diplomacy to educate people must be initiated to inculcate the importance of science and technology in collaboration with other nations. The Counter Terrorism Department (CTD) in Pakistan, has been entrusted with very important responsibilities such as crime scene investigations, cross-examinations, interrogations, dealing with intelligence and anti-terrorism programs. The Counter Terrorism Department (CTD) in Pakistan, has been entrusted with very important responsibilities such as crime scene investigations, cross-examinations, interrogations, dealing with intelligence and anti-terrorism programs.

In addition to suicide bombing and shooting as a form of harassment, terrorists also engage in a variety of financial crimes such as blackmail, bank fraud, fraud on Phone line, computer fraud, credit-card-fraud, fake investment-schemes, currency schemes, forgery in all forms, and insurance-fraud in an effort to raise money. The majority of these scams are performed through misusing information technology (IT) systems and breaching the infrastructure's security barriers. The financial institutions, banks, and other business owners must first install the most recent licensed software for their operations as well as that for network security in order to prevent such incidents. The management of a bank's dedicated intranet and extranet, departmental communications, and interactions with stockholders and stakeholders are all suggested. The science and technology created expressly to combat terrorism includes methods for either detonating explosives early or preventing their ignition. For general intelligence, law enforcement, or public health-related objectives, the bulk of S&T counterterrorism tools are very helpful. The paper [9] argues that psychological problems and influences cannot be disregarded since they have a role in Pakistani terrorism, terrorist acts against historical, geopolitical, and societal targets, and terrorism in general. The outcomes might potentially be used as a road plan to fix this core issue. The

concept of ritual killing, martyrdom, and self-sacrifice is present in the majority of religions, and terrorists use it to persuade their suicide bombers, attackers, and shooters. In actuality, this is a form of mental illness. Though "self-sacrifice" is "haram" in Islam.

Extremism in its different forms is one of the most prominent aspects of terrorism. The National Counter Terrorism Authority (NACTA) in Pakistan is working to eradicate and combat terrorism and extremism through awareness-raising campaigns and other appropriate ways. A helpful document about Pakistan's narrative in relation to terrorism and extremism was presented by NACTA in [7]. It discusses a popular FATWA signed by 1800 renowned muslin religious scholars of all sects of Islam including scholars of Al-Azhar University and Imam-e-Kaba. It refocuses the key elements of the Quran and the Sunnah to promote harmony among all Islamic religious sects. The educational institutions must be closely monitored to see if they are promoting extremism, terrorism, violence, or militancy. If they are, they must be reported to law authorities for prosecution. Violent extremism has many distinct definitions and interpretations; it is a complex problem that is discussed in academic settings. It is perceived and tolerated differently in many groups, based on their unique structures. Academic institutions should not support, advocate, or encourage hate of Pakistan or provide ongoing terrorism training. No person, group, or province may proclaim jihad in the presence of the Pakistan Army, Pakistan Air Force, or Pakistan Navy.

Department of Automation at Shanghai Jiao Tong University in Shanghai, China, created monitoring and surveillance technologies that rely entirely on automated detection and analysis. Based on "Multi-Stream 3D latent feature clustering for abnormality detection in videos" connected with CCTV at key locations, the full systems have been published and documented in [2] and [3]. The "Multi-Level Two Stream Fusion based Spatio-temporal Attention Model for Violence Detection and Localization" is used by the second system. These devices might be deployed in barren locations. how deviant behavior develops.

2 Pakistan Situation In New Wave Of Terrorism

Pakistan in the past several years is facing terrorism and heavily suffered in terms of more than one hundred thousand humans and resources at the cost of Pakistan’s economy. The occupation of Afghanistan by two big powers, resulted in large number of refugees, who had to come to Pakistan. The last two-year extension of the Proof of Registration cards, which were given to Afghan refugees in Pakistan in 2006, occurred in 2021. Moreover, 1.3 million of the 3.7 million Afghans living in Pakistan as per statement of United Nation’s High Commissioner for Refugees. According to the investigations of terrorist activities, it has been found that in majority of foreign elements involved are of afghan origin most having forged national identity card and passport of Pakistan.

Pakistan and Afghanistan have had a protracted and complicated relationship in the past. Geographically speaking, they are neighbors and share the Durand Line, a porous border that stretches over 2,600 km and has been a point of contention between the two nations for decades. A multidisciplinary approach is

frequently necessary for effective counterterrorism measures, comprising coordination between numerous agencies, intelligence sharing, international cooperation, and all-encompassing policies and plans.

To strike a balance between security requirements and individual rights and privacy concerns, it is also important to carefully analyses the ethical and legal issues surrounding the use of these technologies. To improve security, intelligence, and counterterrorism activities, electronic technology can be employed against terrorism in a number of ways. The application of electronic innovations discussed in this research paper have shown to be successful in the area of combating terrorism. The Taliban took control of Kabul in August 2021, and the country's political landscape has shifted dramatically. Pakistan's role in the new Afghan government, its stance on various issues, and its relations with the Taliban and the international community were likely to evolve, but unfortunately, it didn't happen. It is also important to indicate some salient features of relationship and partnership between Pakistan and Afghanistan. A few important issues are discussed next.

- Border Disputes: Pakistan and Afghanistan's shared border has long been a source of friction. Afghanistan claims portions of Pakistani land as its own and has never publicly acknowledged the Durand Line as its boundary with Pakistan. Over the years, this conflict has damaged the relationship.
- Since the Russian invasion of Afghanistan in the 1980s, Pakistan has been sheltering millions of Afghan refugees. This has had both advantages and disadvantages.
- Security Concerns: According to Afghanistan, Pakistan supports and shields elements that challenge the Afghan government, including the Afghan Taliban and other insurgent groups. These accusations have been consistently refuted by Pakistan, which has asserted that it is devoted to promoting stability and peace in Afghanistan.
- Commercial and Trade Relations: Because both nations share a border and strong cultural links, there is potential for further commercial collaboration. The ongoing conflict in Afghanistan and security concerns have, however, hindered trade cooperation.
- Peace Process: Pakistan had helped in the past to arrange negotiations between the Taliban and the Afghan government. These initiatives, which include the intra-Afghan conversation held in Doha, Qatar,
- U.S. Involvement: Pakistan-Afghanistan ties have been impacted by the presence of international forces in Afghanistan, particularly those from the United States and its allies. Pakistan has participated in the U.S.-led efforts to stabilize Afghanistan while also being the subject of criticism.
- Cultural and People-to-People linkages: Despite political and security difficulties, there are strong linkages between the Afghan and Pakistani people on a cultural and familial level. Commonalities in language, culture, and religion establish connections that go beyond politics.

- Building a border barrier along the Durand Line: In an effort to improve security and regulate the flow of people and products across the border with Afghanistan, Pakistan started building a border fence along that country's western border. The fence is a component of a larger border control system intended to stop unlawful cross-border activity like smuggling and militant movement.
- Disagreement and Controversy: Afghanistan has vehemently opposed the building of the fence. The Durand Line itself is one of the major issues in dispute. The Durand Line has never been legally acknowledged by Afghanistan as an international boundary, and it still claims a portion of Pakistani territory as its own. Afghanistan therefore views the building of the fence is not in their interest.

It's crucial to note that since my previous update in September 2021, a lot has changed in Afghanistan. In August 2021, the Taliban seized control of Kabul, and since then, the political climate of the nation has drastically changed. Pakistan's position on many topics and its part in the new Afghan government.

3 Electronic Devices

To improve security, intelligence, and counter-terrorism activities, electronic technology can be employed against terrorism in a number of ways. The following electronic innovations have shown to be successful given in Table 1.

Table 1 : Important Electronic Technologies and Methods to combat with Terrorism

Sr#	Technology and Methods
1.	Artificial Intelligence (AI)
2.	Analytics Drones
3.	Behavioral analytics
4.	Big Data
5.	(CCTV)
6.	Chemical and biological sensors
7.	Cryptography
8.	Explosives Screening & Detection
9.	Communication & Encryption
10.	Fingerprint & iris scanners
11.	Geographic Information Systems
12.	Psychological Profiling
13.	Protecting Critical Infrastructure
14.	Secure Communication Tools
15.	Satellite Imaging
16	Surveillance & reconnaissance
17.	Signal Interception
18.	Social Media's Monitoring
19.	Threat intelligence` Processing
20.	Technology based on biometrics`

4 Recommendations For Using The Technology

In this Section, we highlight a few methods from Table1, which are effective for detection and control of terrorism. The Technology for surveillance and reconnaissance using unmanned Aerial Vehicles (UAVs or called Drones is effective. The Drones are used for intelligence gathering, surveillance, and monitoring in remote places to look for and detect terrorist activities; particularly, while using the satellite imaging method. Counter-drone technology has become crucial for security as terrorists utilize drones more frequently for reconnaissance and potentially for attacks. It is useful to monitor the movements and activities

of the terrorists used with high-resolution satellite imaging as source of intelligence.

However, the employment of closed-circuit television (CCTV) is effective to observe terrorist activity, which can be used with the aid of surveillance cameras placed in public areas, transit hubs, and is therefore, is of vital infrastructure value. The methods based on biometrics facilitate the facial recognition particularly at airports, border crossings, and other high-security places, people can be recognized using facial recognition technology.

The identification of criminal and terrorists is carried out using Fingerprint and iris scanners: These tools are used to verify identities and follow terrorists who have been identified. The algorithms of artificial intelligence and data analytics are used to analyze information from a variety of sources to spot trends and potential dangers. The techniques of Big Data Analysis are used to find patterns and linkages within terrorist networks by analyzing “big datasets”. It is the need of day to observe and monitor closely the Social Media and on media sites for indications of radicalization and terrorist activity. To stop cyberterrorism, it is crucial to ensure the security of vital infrastructure including power grids, transportation networks, and financial institutions.

Geographical Information Systems helps in mapping and analyzing geographic data, which can be critical in understanding and responding to terrorism-related incidents. Also the communications of terrorist can be monitored and jammed by means of encryption breaking techniques. The intelligence and law enforcement agencies extensively use the cryptography to code and decode their messages and confidential reports. The technology for the detection of explosives and dangerous materials is of great significance. The sensors are used to identify chemical or biological dangers. The X-Rays is used for this purpose. One method of identifying prospective terrorists is through Psychological Profiling. Firewalls and Intrusion Detection Systems (IDS): These technologies protect government and critical infrastructure networks from cyberattacks.

According to [4], the other cutting-edge technology known as geo-fencing has shown to be quite successful in identifying criminals and terrorists. As demonstrated below from [4], it functions in tandem and combination with RFID (Radio-Frequency Identification), CRD (Call Record Data), GPS (Global Positioning System), and Wi-Fi. Firewalls and Intrusion Detection Systems (IDS): These technologies protect government and critical infrastructure networks from cyberattacks. The technique is illustrated figure 1. To trigger and activate a marketing action to a mobile device (perhaps a mobile phone), a computer-generated, simulated, cybernetic, and virtual geographic boundary is created. When a user enters or exits this virtual boundary surrounding a certain area, GPS or RFID is detected.

As strongly advised in [6], the employment of cutting-edge technology known as "Demilitarized Zone". This technology gives gangsters no chance, who break into a network access while adding an extra degree of security to the LAN (local area network). Its main objective is to gain access to unreliable networks by intelligence agencies, if implemented successfully.

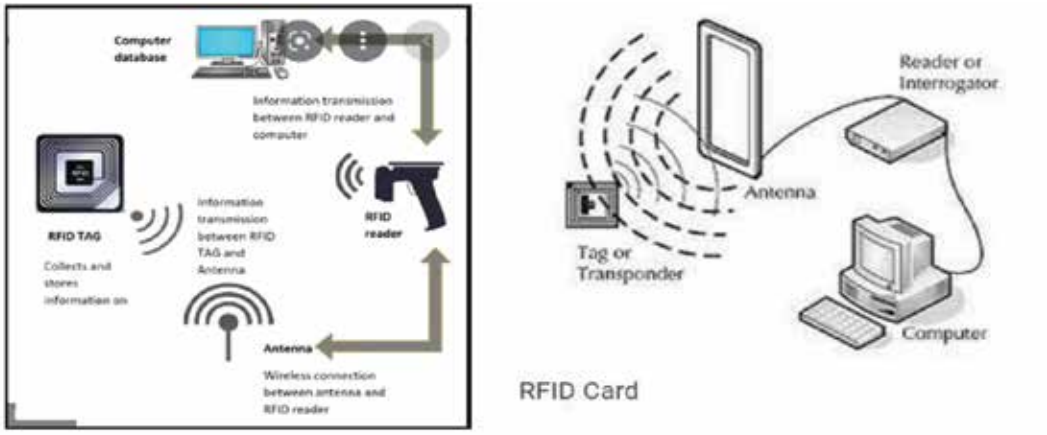

Fig 1: Geo-Fencing Network

For the security and sensitivity of their data, corporations keep data from external sources such as "Voice over Internet Protocol (VoIP)", Domain Name System (DNS), FTP (File Transfer Protocol), Mail, proxies, and their web servers in the Demilitarized Zone. Hackers and trackers find it challenging to obtain the crucial information held by the company because of the **Demilitarized Zone**. To guard against hackers and trackers, it uses two fire walls (the hardware firewall and the software firewall). With courtesy, from [6], the following Demilitarized Zone framework is displayed in Figure 2.

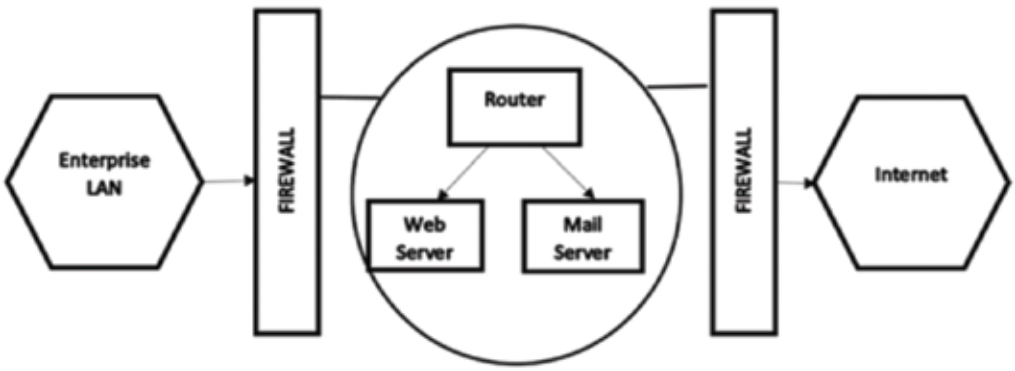

Fig 2: Working of Demilitarized Zone courtesy [6]

5 Acknowledgement

The authors are grateful to Mr Kaukab Jamal Zuberi, the Director and HoD, Department of Criminology and Forensic Sciences and Chief Editor Dr. Syeda Mona Hassan PhD; MPhil; MSc for positive guideline.

6 Conclusion

1. All illegal foreigners including Afghan migrants must be repatriated.
2. To improve economic conditions in the country, smuggling of all commodities in the name of Afghan transit trade must be stopped.
3. Strict measures to combat terrorism are most essentially to be taken.
4. The entry of all foreigners without legal documents must be strictly banned.

7 References

[1] S. Ahmad. "Combating Terrorism Through Technology in Pakistan",CS-RC Centre for Strategic and Contemporary Research. 2022.

[2] M. Asad, H. Jiang, J. Yang, E. Tu, and A. A. Malik. Multi-Stream 3D latent feature clustering for abnormality detection in videos. Applied Intelligence, PP 1-18. 2021.

[3] M. Asad, H. Jiang, J. Yang, A. A. Malik. "Multi-level Two Stream Fusion based Spatio-temporal Attention Model for Violence Detection and Localization", International Journal of Pattern Recognition and Artificial Intelligence. 2021.

[4] A. A. Malik, M. Asad and W. Azeem, "Child Kidnapping and Abuse by Gang-Criminals and the Legitimate Custody of Minor to Parents after Rescue and Use of Geo-fencing to Arrest the Absconding Criminals", International Journal for Electronic Crime Investigation. 6(1):2022.

[5] C. Zdanowicz, C. Alvarado and K. McCleary, "Texas Shooting", CNN. 2023.

[6] A. A. Malik, M. Asad and W. Azeem, "Frauds in Banking and Entrepreneurs by Electronic Devices and Combating Using Software and Employment of Demilitrized Zone in the Networks". International Journal for Electronic-Crime Investigation, Vol. 6 issue 4, 2022.

[7] NACTA," Pakistan's National Narrative against Terrorism and Extremism, Developed & Maintained by IT Wing (NACTA), 2023. https://nacta.gov-.pk/pakistans-national-narrative-against-terrorism-and-extremism/

[8] J. Ebner," Fighting International Terrorism with Social Science Knowledge", Footnotes; Public Information Office. 2005

[9] A. Tamizuddin and T. Mahmood, "Terrorism in Pakistan: the psychosocial context and why it matters", B.J-Psych International 15(1) pp. 20-22. 2018.

[10] United Nation Security Council-Counter-Terrorism Committee (CTC) https://www.un.org/securitycouncil/ctc/news/cted-leads-united-nations-countering-terrorist-travel-programme-national-consultation-mission. 2018.

Khurram *et al. (IJECI)* 2023

(IJECI)
ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for
Electronic Crime Investigation

DOI: https://doi.org/10.54692/ijeci.2023.0703157

Research Article

Vol. 7 issue 3 Jul-Sep 2023

Detection of Malicious software and control using Artificial Intelligence

Syed Khurram Hassan[1] and Muhammad Asif Ibrahim[2]

[1] Institute of Quality and Technology Management, University of the Punjab, Lahore, Pakistan.

[2] Department of Mathematics, FC College University, Lahore.

Corresponding author: khuramshah6515@gmail.com

Received: June 15, 2023; **Accepted**: August 17,2023; **Published**: September 20, 2023

Abstract:

Due to growing pervasiveness of malware in the contemporary digital environment, it is imperative to establish efficient identification and management systems to protect computer systems and networks. Conventional approaches to malware detection frequently encounter difficulties in keeping pace with the swiftly progressing characteristics of malevolent software. However, the emergence of artificial intelligence (AI) has unlocked fresh prospects for augmenting malware detection and control. This article investigates the implementation of AI methodologies in malware detection and mitigation, elucidating the benefits and obstacles connected with this strategy.

Keywords: Malware, virus, worm, trojan, ransomware, spyware.

1. Introduction

In the interconnected digital realm of today, cybersecurity threats are ever-evolving, posing substantial risks to computer systems, networks, and sensitive information. Among these threats, malicious software (malware) stands as a prominent adversary. Malware, commonly referred to as "malicious software," encompasses a diverse category of software programs intentionally designed to cause harm. This article serves as an introduction to malware, providing a comprehensive definition and exploring the various types of malware that exist.

1.1 Defining Malware

Malware comprises a broad range of harmful software programs, each with unique characteristics and objectives. Essentially, malware encompasses any software code or program developed with malicious intent to compro-

mise the integrity, availability, or confidentiality of computer systems and networks. Malware deviates from legitimate software as it aims to infiltrate, disrupt, or damage the target systems it infects [1].

1.2 Types of Malware

1.2.1 Viruses

One of the most renowned forms of malware is the computer virus. Analogous to their biological namesakes, computer viruses propagate and spread by attaching themselves to clean files or programs. Upon execution of an infected file or program, the virus activates and initiates a range of malicious activities, such as data corruption, system impairment, or unauthorized access. Human interaction, such as opening infected email attachments or downloading compromised files from the internet, often serves as the vehicle for virus propagation [2].

1.2.2 Worms

Worms, in contrast to viruses, possess self-replicating capabilities and can spread independently without requiring user interaction. These self-contained programs exploit security vulnerabilities in computer systems or networks, enabling them to propagate from one device to another. Unlike viruses, worms do not necessitate host file attachment. Instead, they replicate and disseminate themselves directly across networks, resulting in widespread damage and excessive consumption of network resources. Worms propagate rapidly, making them particularly perilous to large-scale networks [3].

1.2.3 Trojans

Trojans, also known as Trojan horses, represent deceptive malware programs that masquerade as legitimate software or files, deceiving users into executing them. Unlike viruses or worms, Trojans do not replicate autonomously. Instead, they grant unauthorized access to the user's computer, allowing attackers to engage in various malicious activities. Trojans may establish backdoors, pilfer sensitive information, or install additional malware onto the infected system. They commonly propagate through email attachments, malicious downloads, or compromised websites [4].

1.2.4 Ransomware

In recent years, ransomware has emerged as a prevalent and highly detrimental form of malware. This form of malicious software encrypts the files or entire system of a targeted individual, making them inaccessible until a ransom is provided to the person responsible. Ransomware frequently proliferates through email phishing campaigns, malicious downloads, or exploitation of software vulnerabilities. The financial motivation behind ransomware attacks has turned it into a lucrative tool for cybercriminals, targeting individuals, businesses, and even critical infrastructure [5].

1.2.5 Spyware

Spyware, a clandestine malware variant, covertly collects information about a user's activities without their knowledge or consent. It monitors online behavior, captures keystrokes, records browsing habits, and may even exfiltrate sensitive data such as login

credentials or financial information. Spyware typically operates stealthily, making its detection challenging. Distribution channels for spyware include malicious downloads, infected websites, or bundling with seemingly legitimate software [6].

1.2.6 Adware

While not as malicious as other forms of malware, adware is an intrusive software that inundates users with unwanted advertisements. Adware is often bundled with free software or downloads, with its primary purpose being revenue generation for the creators through targeted advertising. However, adware can consume system resources, impede computer performance, and compromise user privacy by collecting browsing habits and personal information [7].

Malware Type	Replication Method	Spread Without User Interaction	Objectives	Distribution Methods
Viruses	Attaches to files and programs	No	Data corruption, system damage, unauthorized access	Infected email attachments, compromised downloads
Worms	Self-replicating	Yes	Network propagation, resource consumption	Exploiting vulnerabilities, malicious links
Trojans	Disguised as legitimate software	Yes	Unauthorized access, information theft, system compromise	Email attachments, malicious downloads
Ransomware	Exploits Vulnerabilities	Yes	File Encryption, ransom demands	Email phishing, malicious downloads
Spyware	Stealthy	Yes	Unauthorized data collection, monitoring user activities	Infected websites, bundled software
Adware	N/A	Yes	Displaying unwanted ads	Bundled with free software downloads

Table 1: Differentiation of Malware

2 Ai-Based Malware Detection

2.1 IDS/IPS

Intrusion detection and prevention systems utilize an advanced mechanism to constantly monitor the network and detect potential security breaches. These systems maintain a log of pertinent information, address any issues that arise, and promptly alert security administrators. The functionalities of intrusion detection and prevention systems encompass various aspects, including sending notifications to administrators, discarding malicious packets, blocking undesirable network traffic originating from suspicious sources, terminating suspicious connections, and automatically adjusting configurations to counteract future intrusion attempts. There are numerous variations of intrusion detection and prevention, such as network intrusion prevention, host intrusion prevention, network behavior analysis, and wireless intrusion prevention, which can be employed for different applications [8].

2.2 Malware Analysis

The act of identifying and examining malicious software, unwanted entities, and their effects is commonly known as malware

analysis. This process involves uncovering indicators of compromise to detect infected machines, predict future attacks, assess their impacts, and identify compromised systems. Understanding the characteristics and objectives of a suspicious file plays a crucial role in malware detection, and this procedure is referred to as malware analysis. There are various approaches to conducting malware analysis, including static analysis, dynamic analysis, memory analysis, and hybrid analysis, which are used in different operating systems such as Windows, Linux, and Android [9].

Static scrutiny involves extracting static signatures or patterns from binary files without executing them. It is typically considered straightforward and efficient, but it struggles with analyzing obfuscated malware. On the other hand, dynamic malware examination allows malware to execute in an isolated environment, enabling the monitoring of its behaviors. This makes dynamic scrutiny resistant to syntactic obfuscation techniques [10]. However, dynamic scrutiny has limitations in tracking the behaviors of advanced malware like fileless malware. Another approach, memory examination, can reveal malicious behaviors associated with fileless malware [11].

2.3 Static Analysis

Static analysis is a technique used to examine malware without executing it. Its primary goal is to extract metadata from the malware. While static analysis is effective in identifying familiar malware, it faces limitations when dealing with complex and novel malware. Malware creators often employ obfuscation techniques to hide the true nature of their applications. Additionally, they use polymorphism and metamorphism techniques to modify the appearance of the code across different malware samples. Analyzing intricate malware using advanced static analysis approaches is a time-consuming process that requires extensive expertise in operating systems and disassembly. For example, PE Explorer is a tool commonly used to inspect Windows .exe and .dll files. Androguard is a well-known static analysis tool for analyzing Android applications, which facilitates the comparison of code similarities between two applications. By comparing the codes of two applications, Androguard can determine which methods are identical, similar, or present in one but absent in the other [12].

2.4 Dynamic Analysis

Dynamic analysis involves the examination of malware behavior and consequences upon execution. Understanding the actions performed by malware during execution is crucial. The primary goal is to collect real-time information about the behavior of malware and its impact on the system. This approach allows for the comprehensive observation of the malware's functionality and its influence on the surrounding environment during execution. Typically, the file is executed within a virtualized environment. Dynamic analysis is preferred over static analysis because it can detect malware easily, even if the malware's structure undergoes changes, as its behavior and characteristics remain constant. Wireshark

and TCP dump are useful tools for capturing and analyzing network packets. DroidBox is a sandbox tool specifically designed for Android applications. It logs an application's network communications, file accesses, launched services, loaded classes, cryptographic operations using the Android API, messages, and outgoing calls during execution [13].

2.5 Malware Detection Based on Signature

A digital fingerprint is a unique code injected into application software by malicious software creators, serving as a distinguishing identifier for harmful software. It is an efficient and swift method for detecting known malware. However, this technique has limitations when it comes to unfamiliar attacks. It is ineffective in identifying novel and unrecognized malicious software because there is no distinct digital fingerprint available for such attacks. Furthermore, malware developers can constantly modify their code or packaging methods to evade creating an identical digital fingerprint to previous versions, thus circumventing detection [14].

2.6 Malware Detection Based on Behavior

In this method, software behavior is utilized to determine whether it is harmful or benign. A sensor that focuses on behavior goes through three distinct stages:

a. data collection, which involves gathering information about the malware,
b. analysis of the collected data to extract the most relevant details and create a behavioral model or profile,
c. the identification phase which entails finding a correlation between the malware's profile and the one that represents malicious behavior [15].

2.7 Machine and Deep Learning

"In recent years, the fields of research have witnessed significant advancements attributed to the progress made in machine learning (ML) and deep learning (DL). Artificial intelligence has experienced remarkable growth, largely thanks to the contributions of ML and DL. ML, a captivating domain of computer science, has found successful applications in information retrieval, pattern recognition, and decision-making [16].

DL [17], on the other hand, relies on robust and versatile models that facilitate the extraction of relevant information for complex tasks. In this regard, DL holds great potential for the identification, categorization, and analysis of malicious software, as well as the recognition and detection of botnets. It also aids in mitigating cyber attacks, preventing intrusions, responding to incidents, analyzing network traffic, detecting advanced persistent threats (APTs), identifying cybercriminals, conducting thorough packet inspection, and performing analytics for cycybersecurity.

3 DI-based Malware Detection Models In Windows Platform

Jeon and Moon [18] developed an advanced deep learning-based technique for malware detection. Their approach combined static opcode sequences with dynamic recurrent neural networks (RNN) and convolutional recurrent neural networks. To condense

lengthy opcode sequences into concise ones, they employed a convolutional autoencoder. This allowed the recurrent neural network to utilize the opcode features generated by the autoencoder for malware classification. The opcode characteristics were extracted statically from executable files in the Windows environment. The performance of their approach was impressive, achieving a detection accuracy of 96% and a true positive rate of 95%.

To further improve malware detection, Yuan et al. [19] proposed a novel model called MDMC. This model leveraged Markov images and convolutional neural networks (CNN) to identify malware attacks. By utilizing a bytes transfer probability matrix, binary files were transformed into Markov images. The CNN played a crucial role in automatic feature engineering and classification. The experiments were conducted on a Microsoft dataset consisting of 10,868 malware samples, covering nine malware families. The results demonstrated the superior performance of MDMC, achieving an accuracy of 99.264%.

4 DI-based Malware Detection Models In Android Platform

Pektas and Acarman [20] conducted a study on Android application analysis, where they utilized a dataset comprising 25,000 benign and 24,650 malicious applications. Their objective was to develop a deep learning-based model for automatic identification and classification of Android applications. Initially, they employed a Convolutional Neural Network (CNN) to extract relevant features from the applications. The extracted features were then passed to a Long Short-Term Memory (LSTM) module to capture intricate relationships among them. The LSTM module generated a final feature set, which was subsequently input into a dense layer or fully connected neural network for classification. To optimize the hyperparameters of the network, the researchers utilized the grid search approach and implemented the model using TensorFlow and Keras frameworks. Their DL based approach achieved an impressive accuracy of 91.42% in identifying unknown Android malicious applications.

Ma et al. [21] proposed a framework called Droidect for classifying malicious Android applications on Android devices. The framework was based on a Bidirectional LSTM (Bi-LSTM) model. The researchers extracted behavioral features from API call sequences of APK files and applied an NLP-based semantic localization technique to construct dense vectors. These vectors were then fed into the Bi-LSTM model for classification. The evaluation of the framework was performed on a dataset comprising 11,982 benign files and 9,616 malicious files, demonstrating an accuracy of 97.22% for malware detection.

In addition, another study implemented an LSTM-based approach to detect malware in Android applications. The researchers obtained opcode sequences from benign and malware applications, sourced from Play Store and VirusShare respectively. Text processing techniques were employed to preprocess the opcodes, and a LSTM-based malware detector was constructed using Keras. This approach achieved a detection accuracy of 96%. Further-

more, the researchers explored various deep learning-based detectors using LSTM, GRU, Bi-LSTM, and stacked LSTM/GRU models. They utilized permissions, API call sequences, intent sequences, and intent filters for implementing these models [22].

5 DI-based Malware Detection Models In Linux Platform

Xu et al. [23] proposed HawkEye, a system designed for identifying malware attacks in Linux. The system utilizes control flow graphs (CFGs) and employs graph neural networks (GNNs) in combination with a multilayer perceptron-based classifier. To gather malware samples, the researchers accessed the AndroZoo and VirusShare repositories, while benign samples were obtained from executable files and libraries in a clean Ubuntu installation. To represent the structural information of both malware and benign executables, graph sets were defined using a CFG extractor. The CFG extractor captured details such as basic block addresses and assembly instruction/opcodes. These extracted features were then processed by the GNN module to generate graph embedding features, which were subsequently fed into the MLP classification module. The evaluation of the system showcased a remarkable detection accuracy of 96.82% when identifying malware in a Linux environment.

6 Types Of Ai Based Malware Detection

In order to develop a comprehensive comprehension of the diverse AI-based methodologies utilized for malware detection, it is crucial to grasp the fundamental concept of malware itself and its operational mechanisms. Malware, an abbreviation for malicious software, encompasses software that is purposefully crafted to inflict harm or render computer systems inoperable. This category of malicious software encompasses various types such as viruses, worms, Trojans, spyware, and adware. Given the substantial negative consequences caused by malware, extensive research has been devoted to analyzing and countering these malevolent programs. With the recent advancements in Artificial Intelligence (AI), cybersecurity experts have increasingly turned their focus to utilizing Machine Learning (ML) and Deep Learning (DL) techniques to enhance the identification and categorization of malicious files [24].

Recent research findings suggest that individuals with limited experience often encounter difficulties when differentiating between benign and malicious applications. This underscores the importance of designing computer systems and mobile applications that possess the capability to detect malicious activities and safeguard all stakeholders involved. A plethora of algorithms has been developed to identify malware activities, leveraging cutting-edge concepts such as Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) [25]. In the realm of cybersecurity, AI has gained substantial prominence, particularly in the field of malware detection, where AI-based approaches are increasingly prevalent. Malware analysis forms the bedrock of effective malware detection techniques and is imperative in understanding the classification

and functionality of malicious files [26].

Machine learning has gained popularity as an AI-based technique for malware detection. By analyzing patterns in datasets, machine learning algorithms can effectively identify malware, even if it's a previously unseen type. However, it is important to note that machine learning can be computationally demanding and requires a significant amount of training data to optimize the algorithms. On the other hand, deep learning (DL) has shown effectiveness in detecting sophisticated malware that constantly evolves [27].

6.1 Methods

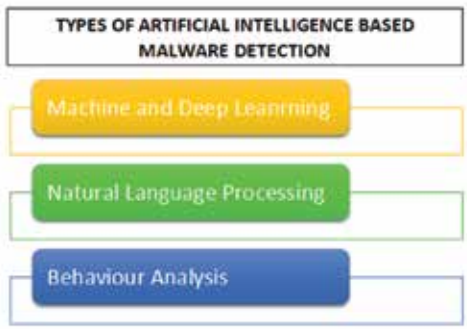

Fig 1: Types of AI based Malware Detection

6.1.1 Machine and Deep Learning

There is a diverse range of AI-based malware detection methods available, each with its own set of advantages and disadvantages. Among the most common methods are machine learning and deep learning. Extensive research has been conducted on utilizing deep learning algorithms for malware detection. Machine learning is an AI technique that can learn from data and improve its performance over time. It finds applications in areas such as spam filtering and fraud detection. On the other hand, deep learning is a more advanced form of machine learning that can learn from data in a manner resembling human learning. It is commonly used for tasks like image recognition and natural language processing. Both machine learning and deep learning have distinct strengths and weaknesses in the context of malware detection. However, it is worth noting that while deep learning approaches have demonstrated impressive results in malware detection, several of these studies have limitations due to either [28].

6.1.2 Natural Language Processing

Natural Language Processing (NLP) is a widely used AI-based method for detecting malware in the field of cybersecurity. NLP focuses on enabling computers to understand and process human language, ranging from basic word recognition to complex tasks like sentence comprehension and information extraction. In the context of malware detection, NLP offers several advantages. It can identify specific keywords and patterns commonly associated with malicious software [29]. For example, instructions to "delete all files" or "format the hard drive" are indicative of potential malware. NLP can also analyze sentence structures to identify suspicious commands such as "download this file" or "install this program." NLP is a powerful tool for detecting various forms of malware. However, it is essential to acknowledge that no single method is foolproof. To achieve optimal effectiveness, NLP should be combined with other malware detection techniques. By integrating NLP with

complementary approaches, security researchers can enhance the overall accuracy and efficacy of malware detection systems. It is important to adopt a multi-faceted approach to ensure comprehensive protection against evolving and sophisticated malware threats [30].

6.1.3 Behaviour Analysis

Email attachments are a common avenue for the distribution of malware, as opening an infected attachment can trigger the execution of malicious code and compromise a computer. Once infected, malware can carry out various nefarious activities, such as file deletion, data theft, or even taking control of the entire system. To combat this threat, many organizations have turned to AI-based malware detection methods. These methods utilize artificial intelligence algorithms to analyze email attachments and other files, with the aim of identifying potential threats. By doing so, they can proactively block emails containing suspicious attachments from reaching users' inboxes. AI-based malware detection methods employ various approaches. Some methods focus on analyzing the content of files to identify potentially malicious code. This involves examining the structure and code of the file for known patterns or indicators of malware. Other methods concentrate on observing program behavior to detect signs of malicious intent. By monitoring program execution and analyzing actions such as file modifications or network communications, these methods can flag potentially malicious programs. Some approaches combine both content-based and behavior-based analyses to achieve improved accuracy and comprehensive coverage. The adoption of AI-based malware detection methods in email security strengthens protection against threats stemming from attachments. By harnessing the capabilities of artificial intelligence, organizations can fortify their defenses by proactively identifying and blocking potentially malicious attachments. This helps safeguard users and systems from the risks associated with malware [31].

6.2 Issues and Challenges

The shortcomings of traditional methods in malware detection and analysis have prompted researchers to seek alternative technologies that can achieve real-time detection with high accuracy and a reduced false positive rate. Earlier approaches relied on statistical analysis of system changes or employed probabilistic methods that looked for specific literals to classify executable as malware. However, these probabilistic and statistical techniques provided only approximate assessments based on a limited set of malware features and encountered challenges when dealing with obfuscated malware [32].

a. The utilization of packed executables and the presence of small datasets can introduce uncertainties in the outcomes when deploying a real-time malware detection solution. These factors can significantly affect the effectiveness and accuracy of the implemented solution, underscoring the importance of developing robust techniques capable of handling such scenarios.

b. When employing a framework that relies on Windows audit logs, the effectiveness of the solution can be compromised by obfuscation techniques. In such instances, the approaches outlined in research papers may not be directly applicable or may require modifications to address the challenges posed by obfuscation. This adaptability is crucial to ensure the solution remains reliable and useful in detecting malware.

The field of malware analysis has witnessed the rise of deep learning as a prominent approach, primarily because of its capacity for automatic feature engineering. However, despite its advancements, there remain certain unresolved issues that demand attention. Notably, deep learning-based methods encounter difficulties when confronted with limited availability of data. This constraint calls for further exploration and research in various critical domains [33].

Addressing these challenges requires ongoing research and development in the field of AI-based malware detection, as well as collaboration between cybersecurity experts, AI researchers, and the wider security community to improve the effectiveness and reliability of these systems. Some suggestions that this paper gives for the betterment in the realm of Artificial Intelligence based Malware Detection are as follows [34]:

i. Comprehensive and Diverse Training Data: Acquiring a wide range of labeled training data that covers various malware types, families, and variants is crucial. It is important to continuously update the training data to account for emerging threats and the evolving landscape of malware.

ii. Integration of Behavioral Analysis: In addition to static file analysis, incorporating behavioral analysis techniques is valuable. By examining the dynamic behavior of programs and identifying anomalies, it becomes possible to detect malicious activities, even in the absence of known malware signatures.

iii. Ensemble Models: Utilizing ensemble models that combine predictions from multiple AI algorithms or models can enhance detection accuracy. Ensemble methods leverage the strengths of different models while mitigating individual model weaknesses, leading to improved overall performance.

iv. Adversarial Training and Testing: Training AI models using adversarial samples can enhance their resilience against adversarial attacks. By exposing models to manipulated or modified malware samples during training, they can learn to detect and counteract adversarial attempts to evade detection.

v. Continuous Learning and Updates: Implementing mechanisms for ongoing learning and updates is vital to keep AI models current with emerging threats. Regularly retraining models using new data and periodically updating detection algorithms and techniques will enable systems to stay

ahead of evolving malware tactics.

7 Traditional Methods

7.1 Signature-Based Detection

Signature-based detection is one of the most prevalent techniques used in traditional antimalware systems. It relies on the identification of known malware patterns, referred to as signatures, which are stored in databases. When a file is scanned, its content is compared against these signatures. If a match is found, the file is flagged as malware and appropriate actions are taken [35].

The strengths of signature-based detection lie in its effectiveness against known malware strains and its efficiency in rapidly identifying threats. By leveraging a vast database of signatures, it can quickly identify and quarantine files that exhibit known malicious patterns. This approach has been refined over the years, enabling anti-malware software to keep pace with the constantly evolving threat landscape. However, its primary limitation is the inability to detect novel or modified malware that does not match existing signatures. Attackers can easily evade signature-based detection through techniques like polymorphism or code obfuscation, which alter the characteristics of malware without changing its underlying functionality. To mitigate these limitations, heuristics and behavioural analysis techniques are often combined with signature-based detection to enhance the detection capabilities of anti-malware solutions.

7.2 Behavioural Analysis

Behavioural analysis focuses on observing and monitoring the behaviour of files, programs, or processes to identify potential malware. This approach aims to detect malicious activity by examining actions such as system modifications, unauthorized network communications, or attempts to exploit vulnerabilities.The advantage of behavioural analysis is its capability to detect previously unknown malware or variants that have undergone modification. By analysing the behaviour of software, it can identify suspicious activities and anomalies indicative of malware. For example, if a program attempts to modify critical system files or establish connections with suspicious external servers, it raises red flags and triggers appropriate response measures. However, this technique may generate false positives due to legitimate software exhibiting similar behaviour or false negatives if malware remains dormant during analysis. Additionally, behavioural analysis can be resource-intensive, requiring the continuous monitoring of system activities and the establishment of baselines for normal behaviour. To address these challenges, machine learning algorithms and anomaly detection techniques are often employed in behavioural analysis. These approaches leverage historical data and behavioural patterns to identify deviations from normal activities, enhancing the accuracy of malware detection [36].

8 Limitations Of Artificial Intelligence In Malware Detection

Due to the rise in malware activity brought on by the quick development of technology, security has grown to be a significant concern and now threatens the safety and security of

both computer systems and stakeholders. One of the most urgent concerns is safeguarding the data from fraudulent attempts in order to preserve stakeholder security, notably that of end users. Malware is a collection of harmful programming code, scripts, active content, or intrusive software that is meant to damage legitimate computer programmers, mobile, or web apps [37].

A study found that novice users can't tell the difference between perilous and trustworthy programmer. Therefore, malicious activity detection should be built into computer systems and mobile applications to safeguard stakeholders. There are several techniques to identify malware activity that make use of cutting-edge ideas like artificial intelligence, machine learning, and deep learning. In this study, we focus on AI-based strategies for identifying and thwarting malware activity [38].

Cyberattacks are increasingly using machine learning (ML) and artificial intelligence (AI) techniques. AI aids in the creation of hidden channels and the malware's concealment. AI also facilitates difficult-to-detect cyber-physical sabotage and new varieties of phishing attempts. Malware developers are increasingly using AI and ML techniques to enhance the effectiveness of their attacks [39].

Defenders must consequently prepare for unusual malware with cutting-edge, evolving features and functionalities. The ability of AI to automate difficult operations poses a difficulty in the face of the defensive application of anti-malware AI techniques. This article reviews the current state of evasion and attack methods used by AI-enhanced malware against AI-supported defense systems [40].

8.1 AI-Based Real-Time Malware Detection in Data Centers

To enhance security in Data Centers (DCs) and Smart Cities (SCs), an AI-powered edge computing approach called pAElla has been developed. pAElla utilizes real-time malware detection (MD) on an IoT-based monitoring system for DCs/SCs. By analyzing power measurements' spectral density and employing autoencoders, pAElla achieves promising results with a high F1-score and low false alarm and malware miss rates [41, 42].

8.2 False Positive/Negative Rate

There may be some similarities between the fingerprints and characteristics of malicious files and samples. False positive and false negative rates are a problem for a number of malware detection methods. However, an expansion in misleading positive or bogus negative rates diminishes the model recognition precision, misleading up-sides are definitely huger than misleading negatives in the powerful malware identification models. On a user's computer, if a legitimate file is mistakenly identified as malicious, the operating system and other applications may cease to function [43].

8.3 Insufficient Training Data

AI models, particularly those based on machine learning, heavily rely on extensive and diverse training datasets to learn patterns and make accurate predictions. However,

acquiring comprehensive and up-to-date labeled training data for malware detection presents a challenge. The constantly evolving nature of malware makes it difficult to maintain training datasets that are current and representative of the wide range of threats [44].

To mitigate this limitation, researchers are exploring techniques such as data augmentation, transfer learning, and active learning [45]. Data augmentation involves generating synthetic malware samples or manipulating existing samples to expand the training set. Transfer learning leverages knowledge from pre-trained models on related tasks to enhance the performance of malware detection models. Active learning methods prioritize the selection of the most informative samples for manual labeling, optimizing the use of limited resources and improving the training dataset [46, 47].

To overcome this limitation, AI-based malware detection systems are often supplemented with other techniques such as behavior monitoring, heuristics, and anomaly detection. These methods focus on identifying suspicious behaviors or deviations from expected patterns, providing an additional layer of defense against zero-day attacks. Collaboration and information sharing among security communities are also crucial in rapidly detecting and responding to emerging threats [48].

8.4 Interpretability and Explainability

Many AI models, especially deep learning models, are often considered black boxes, lacking interpretability and explain ability. These models operate through complex mathematical computations, making it challenging to understand the underlying reasons behind their decisions. In the context of malware detection, this lack of transparency can be problematic, as it becomes difficult for security analysts to trust and validate the alerts generated by AI models [49].

Researchers are actively working on developing methods for interpreting and explaining AI models' decisions in the context of malware detection. Techniques such as model-agnostic methods, attention mechanisms, and rule extraction algorithms aim to provide insights into the decision-making process of the model. By gaining understanding of the factors and features that contribute to the model's predictions, security analysts can gain more confidence in the alerts generated by AI-based malware detection systems [50].

8.5 Resource Requirements

Some AI models used in malware detection, particularly deep learning models, can be computationally expensive and demand significant computational resources. Deploying such resource-intensive models on devices or networks with limited resources can pose challenges, impacting the performance and scalability of the malware detection system [51].

To overcome this limitation, researchers explore techniques such as model compression, pruning, and hardware acceleration. Model compression methods aim to reduce the size and complexity of the AI model without

significantly compromising performance. Pruning techniques remove unnecessary connections or parameters from the model, reducing computational requirements. Hardware acceleration, such as utilizing specialized hardware like GPUs or dedicated AI accelerators, can expedite the inference process and improve efficiency [52].

By optimizing resource utilization, researchers strive to make AI-based malware detection systems more accessible and practical for deployment across various platforms and environments [53].

9 REFERENCES

[1] Tahir, R. A study on malware and malware detection techniques. International Journal of Education and Management Engineering, vol. 8, no. 2, 2018.

[2] Molina-Coronado, Borja; Mori, Usue; Mendiburu, Alexander; Miguel-Alonso, Jose (1 January 2023). "Towards a fair comparison and realistic evaluation framework of android malware detectors based on static analysis and machine learning". Computers & Security. vol. 8, no. 1, 2018.

[3] Peter Szor. The Art of Computer Virus Research and Defense. Pearson Education. p. 204. 2005.

[4] Landwehr, C. E; A. R Bull; J. P McDermott; W. S Choi. A taxonomy of computer program security flaws. DTIC Document. 2012.

[5] Richardson, Ronny; North, Max "Ransomware: Evolution, Mitigation and Prevention". International Management Review. 13 (1): 10–21. 2019.

[6] Russinovich, Mark "Sony, Rootkits and Digital Rights Management Gone Too Far". Mark's Blog. Microsoft MSDN. 2009.

[7] Casey, Henry T. "Latest adware disables antivirus software". Tom's Guide. Yahoo.com. Archived from the original on 27 November 2015. Retrieved 25 November 2015.

[8] M. Korcak, J. Lámer, F. Jakab, Intrusion Prevention/Intrusion Detection System (IPS/IDS) for Wifi Networks. International journal of Computer Networks & Communications. 2016.

[9] P. Maniriho, A. N. Mahmood, M. J. M. Chowdhury, A study on malicious software behaviour analysis and detection techniques: Taxonomy,current trends and challenges, Future Generation Computer Systems 130. Pp. 1–18. 2022.

[10] F. Biondi, T. Given-Wilson, A. Legay, C. Puodzius, J. Quilbeuf, Tutorial: An overview of malw Leveraging Applications of Formal Methods,Springer, pp. 565–586. 2018.

[11] R. Sihwail, K. Omar, K. A. Z. Ariffin,

An effective memory analysis for malware detection and classification, CMC-Computers Materials &Continua. Vol. 67, no. 2. pp 2301–2320. 2021.

[12] R. Jusoh, A. Firdaus, S. Anwar, M. Z. Osman, M. F. Darmawan, M. Faisal, Malware Detection Using Static Analysis in Android: a review of FeCO (Features, Classification, and Obfuscation). 2021

[13] O. Or-Meir, N. Nissim, Y. Elovici, L. Rokach, Dynamic Malware Analysis in the Modern Era—A State of the Art Survey. 2005.

[14] A. M. Abiola, M. F. Marhusin, Signature-Based Malware Detection Using Sequences of N-grams. International Journal of Engineering and Technology (UAE). 2004.

[15] H. S. Galal, Behavior-based features model for malware detection. Journal of Computer Virology and Hacking TecTechniques. 2004.

[16] Bernardi, L.; Mavridis, T.; Estevez, P. 150 successful machine learning models: 6 lessons learned at booking.com. roceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, New York, NY, USA, 4–8 August 2019.

[17] Zhu, D.; Jin, H.; Yang, Y.; Wu, D.; Chen, W. DeepFlow: Deep learning-based malware detection by mining Android application for abnormal usage of sensitive data. In Proceedings of the IEEE Symposium on Computers and Communications, Heraklion, Greece, 3–6 July 2017.

[18] S. Jeon, J. Moon, Malware-detection method with a convolutional recurrent neural network using opcode sequences, Information Sciences. Vol. 535. pp 1–15. 2020.

[19] B. Yuan, J. Wang, D. Liu, W. Guo, P. Wu, X. Bao, Byte-level malware classification based on markov images and deep learning, Computers &Security. Vol. 92. 2020.

[20] A. Pektas¸, T. Acarman, Learning to detect android malware via opcode sequences, Neurocomputing. Vol. 396. pp 599–608. 2020.

[21] Z. Ma, H. Ge, Z. Wang, Y. Liu, X. Liu, Droidetec: Android malware detection and malicious code localization through deep learning, arXiv preprint arXiv: 2002.

[22] R. Feng, J. Q. Lim, S. Chen, S.-W. Lin, Y. Liu, Seqmobile: An efficient sequence-based malware detection system using rnn on mobile devices, in: 2020 25th International Conference on Engineering of Complex Computer Systems (ICECCS), IEEE, pp. 63–72. 2020

[23] P. Xu, Y. Zhang, C. Eckert, A. Zarras, Hawkeye: cross-platform malware detection with representation learning on graphs, in: International Conference on Artificial Neural Networks, Springer, pp. 127–138. 2021.

[24] Marais B., Quertier T., Morucci S., AI-based Malware and Ransomware Detection Models. arXiv:2207.02108 [cs.CR], 2022.

[25] James O., Employing Artificial Intelligence Techniques for the Detection and Prevention of Malware, 2022.

[26] Wolsey A., The State-of-the-Art in AI-Based Malware Detection Techniques: A Review. arXiv:2210.11239 [cs.CR], 2022.

[27] Austin B., Maanak G., Senior M., IEEE, Mahmoud A., Automated Machine Learning for Deep Learning Based on Malware Detection, 2023.

[28] Umm-e-Hani T., Faiza B., Muhammad H., Asifullah K., Yeon S., A Survey on the Recent Trends in Deep Learning Based Malware Detection, 2022

[29] A. Vaswani et al., "Attention Is All You Need," in Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, pp. 5998-6008. 2017. Available: https://arxiv.org/abs/1706.03762

[30] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), Minneapolis, MN, USA, pp. 4171-4186. 2019. Available: https://arxiv.org/abs/1810.04805

[31] V. Ramanathan et al., "Deep Behavior Mining: Discovering Unusual Crowd Activities in Videos," in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 36, no. 5, pp. 898-910, 2014.

[32] A. Faitouri, Z. Anazaida, A. Fuad, A. Bander, A. Taiseer, and A. Asma, "Malware Detection Issues, Challenges, and Future Directions: A Survey," 2022.

[33] M. J. Hossain Faruk et al., "Malware detection and prevention using artificial intelligence techniques," in 2021 IEEE International Conference on Big Data (Big Data), IEEE, 2021.

[34] A. Libri, A. Bartolini, and L. Benini, "pAElla: Edge AI-based real-time malware detection in data centers," IEEE Internet of Things Journal, vol. 7, no. 10, pp. 9589-9599, 2020.

[35] Christodorescu, M., Jha, S., & Song, D. Malware detection using behavioral analysis. ACM Conference on Computer and Communications Security (CCS),

Alexandria, VA, USA. 2002.

[36] Szor, P. The Art of Computer Virus Research and Defense. Addison-Wesley Professional. 2001.

[37] A. Mishra, Y. I. Alzoubi, M. J. Anwar, and A. Q. Gill, "Attributes impacting cybersecurity policy development: An evidence from seven nations," in *Proceedings of the IEEE International Conference on Big Data (Big Data)*, pp. 2982-2987. 2021.

[38] M. J. Hossain et al., "Malware detection and prevention using artificial intelligence techniques," in *Proceedings of the 2021 IEEE International Conference on Big Data (Big Data)*, pp. 4845-4850. 2021.

[39] R. Faruk et al., "Detection of cyber attacks using machine learning," in *AIP Conference Proceedings*, vol. 2405, no. 1, 2022.

[40] A. Djenna et al., "Artificial Intelligence-Based Malware Detection, Analysis, and Mitigation," in *Proceedings of the International Conference on Information Science and Systems (ICISS)*, 2022.

[41] A. Libri, A. Bartolini, and L. Benini, "pAElla: Edge AI-based real-time malware detection in data centers," in *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9589-9599, 2020.

[42] N. Muchammad et al., "Malware Detection: Issues and Challenges," in *Journal of Physics: Conference Series*, vol. 1807, no. 1, p. 12-31. 2021.

[43] A. Ribeiro, S. Singh, and C. Guestrin, ""Why should I trust you?" Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135-1144. 2016.

[44] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," arXiv preprint arXiv:1605.07277, 2016.

[45] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," arXiv preprint arXiv:1510.00149, 2015.

[46] C. Zhang et al., "Optimizing convolutional neural networks for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 1-9. 2015.

[47] A. M. Saxe et al., "On the information bottleneck theory of deep learning," in *Journal of Statistical Physics*, vol. 177, no. 3-4, pp. 718-751, 2019.

[48] C. Kolias et al., "DDoS in the IoT: Mirai

and other botnets," in *Computer*, vol. 50, no. 7, pp. 80-84, 2017.

[49] K. Rieck et al., "Learning and classification of malware behavior," in *Journal of Machine Learning Research*, vol. 9, pp. 2721-2764. 2008.

[50] M. Guo, G. Wang, H. Hata, and M. A. Babar, "Revenue maximizing markets for zero-day exploits," in *Autonomous Agents and Multi-Agent Systems*, vol. 35, no. 2, pp. 36-51, 2021.

[51] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why should I trust you?" Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135-1144. 2016.

[52] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," arXiv preprint arXiv:1605.07277, 2016.

[53] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," arXiv preprint arXiv:1510.00149, 2015.

Humaira *et al. (IJECI)* 2023

(IJECI)
ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for
Electronic Crime Investigation

DOI: https://doi.org/10.54692/ijeci.2023.0703158

Research Article

Vol. 7 issue 3 Jul-Sep 2023

Malware Attacks Detection in Network Security using Deep Learning Approaches

Humaira Naeem and Asma Batool
Department of Computer Science, Virtual niversity of Pakistan
Corresponding author: humairanaeem@vu.edu.pk

Received: June 20, 2023; **Accepted:** August 20,2023; **Published:** September 20, 2023

Abstract:

This abstract provides an overview of the study on the use of deep learning approaches, specifically Recurrent Neural Networks (RNNs) and Long Short-Term Memory Networks (LSTMs), for detecting malware attacks in network security. The increasing sophistication of malware attacks has made it challenging for traditional signature-based approaches to detect them effectively. Deep learning algorithms offer the potential to address these challenges, as they can automatically learn complex representations of the data and adapt to new and evolving threats. The study focused on the collection and analysis of a large and diverse dataset of both benign and malicious software samples, which were used to train and validate the deep-learning models. The results of the study showed that the RNN and LSTM algorithms outperformed traditional signature-based approaches in terms of accuracy and efficiency in detecting malware attacks. Additionally, developing more efficient and scalable training methods for deep learning algorithms is an important area for future research. Overall, the future of malware detection using deep learning is promising, and continued research in this field holds great potential for improving the security of our digital systems.

1. Introduction

As more and more businesses and organizations rely on networked systems to store and process sensitive information, the threat of malware attacks has become an increasingly pressing concern. Malware, or malicious software, can take many forms, including viruses, Trojans, and worms, and can cause significant damage to both individual computers and entire networks. To combat this threat, researchers and practitioners in the field of network security have developed various methods for detecting and mitigating malware, ranging from signature-based detection to heuristic analysis[1].

In recent years, the field of deep learning has

emerged as a powerful tool for detecting and classifying malware in network security. Deep learning is a subset of machine learning that relies on artificial neural networks to identify patterns in large datasets, and it has been used successfully in a wide range of applications, including image and speech recognition, natural language processing, and even game playing. In the context of malware detection, deep learning models can be trained on large sets of labeled data to identify common features and characteristics of different types of malware and to classify new instances of malware with a high degree of accuracy [2]

The increasing complexity and sophistication of malware attacks have made traditional signature-based approaches to malware detection insufficient. Malware attacks can cause significant harm to individuals, organizations, and society, making the detection of such threats a critical issue in network security. Deep learning, a subfield of artificial intelligence, has shown great promise in addressing these challenges. The ability of deep learning algorithms to automatically learn complex representations of data and adapt to new and evolving threats makes them ideal for detecting malware attacks [3]. Malware is a kind of suspicious software used by cyber thieves to steal data and destroy systems to obtain unauthorized access to the entire system or an individual's account. Criminals accomplish this by sending users emails or files with a link that must be clicked for the virus to be installed. Furthermore, as the number of undiscovered malware threats grows, security measures, particularly in the case of system security, are becoming an increasingly crucial element of our everyday life. Malware has posed a risk to both consumers and businesses. Since then, a large number of distinct malware versions are created to wreak as much damage and inflict as much disruption as possible. Although to mitigate these attacks, a variety of strategies have been developed to prevent malware attacks. Hence, in this paper, a variety of approaches have been studied in-depth with the purpose of better understanding to introduce the best model for detecting malware attacks in network security. The following detection model has been studied in this paper[4]

- The investigation findings into the Attention Residual Network-based Visualization model show that the proposed method for identifying RGB and grayscale images has a greater accuracy rate. [5]

- The Deep Neural Network approach was investigated, in which the dataset was loaded into the CPU's memory, and then the CNN approach was utilized to detect malware attacks, providing a 95% accuracy. [6]

- The Malware identification was done using a Complex-Network-based Approach. MDCN has higher accuracy and fewer FP (False-Positive) cases, according to a study. [7]

- The study of effective run-time development for visual detection of malware using scalability and a hybrid model of deep learning approaches yielded excellent results. [8]

Additionally, the attacks carried out by cybercriminals to compromise the network are depicted in Fig. 1: If preventative precautions are not taken properly, cybercriminals can quickly gain access to any network by using these techniques. So, to secure the network, malware identification is required.[9][10]

Fig 1: Malware Attacks

As previously said, the best analysis for choosing the best malware detection model has been developed after carefully examining about 15 research articles. As a result, the accuracy of deep learning algorithms like RNN and LSTM outperforms that of traditional detection models. Additionally, the outcomes rates can be more precise than those of the earlier research if the suggested methodology is used Furthermore, Fig . 2 vividly illustrates how dangerous malware attacks are by showing them [11][12].

Fig 2: Malware-Attacking Trends

The rest of the article is divided into the following sections: In Section II, a review of the literature is presented. Described in Section III is the Proposed Methodology. The proposed system for evaluation is shown in Section IV. Performance and outcomes are covered in Section V, and the conclusion is provided in Section VI.

2 Related Work

To defend against malware that is harmful to the network. Many studies have been conducted. Several strategies have been put out by the researcher to stop malware attacks on network security. Furthermore, as a result of this, people are more aware of malicious attacks, although there are still gaps that must be closed over time. Additionally, the following earlier papers have been looked at for this research[13].

Authors in [14] mentioned the techniques along with network scanners, anti-virus, and intrusion detection systems inside the community to locate malware that is hard to identify the malware attacks. As a resul t, they proposed malware detection with the use of Complex Network, a complicated network-primarily based malware detection approach that makes use of the software application Interface name Transition Matrix (API-CTM) to generate complicated community topology after which extracts various functions with the aid of studying distinct metrics of the complicated network to differentiate malware and benign applications. Furthermore, this studies well-known shows that MDCN indicates better accuracy to hit upon Malware

with lower fake-fine instances. They also determined that both malware and benign application networks show contained mixing and observe a power-law degree distribution. The MDCN approach can be implemented in large organization networks as a protection degree against polymorphic malware assaults which can be tough to stumble on with current solutions[15].

Malware identification is currently a significant aspect of research in the field of computer security, according to Diangarti Bhalang Tariang[2], as a result of the exponential growth of malware subtypes. Due to the difficulties in reverse-engineering program executables, gathering real-time execution traces, and manually producing efficient feature units, traditional malware detection and classification techniques like static code analysis and dynamic execution evaluation—which are frequently combined with machine mastering—have limitations. He presented forth a technique for categorizing malware that relies solely on the visual representation of malware software binaries and employs an attention residual module to uncover capabilities that are drawn from various CNN levels [16].

There is a range of ways to protect mobile devices against malware penetration, according to Seyed Mehdi Shahidi and his fellow researchers [3], but many of them miss the accuracy needed to detect Trojan infection. In identifying the malware in this study, deep learning techniques including deep neural networks and the group of handling data are used to detect the malware. With improvements of 10.4% and 31.9%, respectively, it reveals that they are capable of producing results that are superior to those obtained using machine learning approaches. The results of adversarial and non-adversarial approaches are superior when compared to those obtained with machine learning-based algorithms like SVM, RF, and KNN [17][18].

According to Gueltoum Bendiab and his team members[19], As more IoT devices and technologies are deployed, malware's complexity and penetration rates have increased, making it a more challenging problem. Lacking sufficient security measures, a significant quantity of sensitive data is exposed to cybercriminals, who can use it to commit several illegal activities. As a result, improved network security systems that can perform run-time traffic assessment and damaging traffic reduction are necessary. To address this issue, they provide a unique internet of things malware traffic assessment technique that uses DP and visual representation to detect and categorize new viruses more quickly (zero-day malware). Based on testing and comparisons with different neural networks, With an overall accuracy of 95.0%, the ResNN50 has proven to be the most effective at recognizing malware network traffic [20].

Paul Prasse along with his collogues describe in their research that to avoid traffic on network monitoring, a growing percentage of malware employs the encrypted HTTPS protocol [21]. They go into the topic of identifying malware

on client devices using HTTPS traffic analysis. Additionally, They also cover a scalable method for building a malware detection methodology and obtaining communication infrastructure from apparently damaging and helpful application training data utilizing an LSTM network and a neural language model. They created and tested an LSTM-based malware detection model that relied solely on observable HTTPS data components for detection [22].

Ping Yan & Zheng Yan explain in their research work that the remarkable advancements of mobile devices encourage their widespread use [23]. As mobile devices become more integrated with unbiased observer apps, new threats and security problems arise. On the other side, present malicious mobile detection and analysis techniques are useless, unproductive, and unsatisfactory. They provide a comprehensive overview of dynamic mobile malware detection in this study. The first section examines mobile malware's definition, development, categorization, and security concerns[24].

Shanxi Li1 and Qingguo Zhou1 show how to identify malware attacks on system software using machine learning algorithms in their research [25]. According to them, owing to the speedy growth of anti-detection technologies, traditional detection methodologies based on static and dynamic analysis have limited effects. AI-based malware detection achieved prominence in the near times due to its improved prediction performance. However, given the variety of malware, extracting features from it is difficult, making malware detection incompatible with AI technology. In addition, they conducted a comparison with different machine learning techniques and the outcomes show that the approach performs greater in the vast majority of detecting scenarios, with a higher precision of 98.32 percent. Furthermore, they also stated that future research will be focused on adaptive model detection using the GCN.

A thorough malware detection system must be developed due to the ongoing risk of zero-day attacks and the enormous increase in the amount of new malware created every day. To detect breaches, Shamika Ganesan claims that contemporary computer security developments have blended AI technology's capacities with employee performance. The use of malware byte information for machine learning-based techniques to better evaluate the malware file has been superseded by the usage of an image-based intrusion detection system. The effectiveness of Residual Attention for malware detection has been evaluated against existing CNN-based approaches and traditional GIST-based Machine Learning methods [26].

Nan Zhang and his colleagues provide a Malware attack detection model for security systems. Malware detection, they claim, is one of the most powerful and effective methods for ensuring security [27]. Learning-based malicious software detection technology for Mobile is always improving. This is an Android malware detection framework that detects malware automatically. The notion of TC-main Droid comes from the field of text classification. They propose a novel

Android malware framework that combines text classification with a convolutional neural network to improve malware detection for Android-based devices in smart cities. They demonstrated TC-Droid, an Android malware detection tool that does not require feature selection by hand. Feature representations that can be detected automatically.

Xiaojie and Hossain Sayyedi highlight the security issues and the detection method for malware attacks on IoT in their research. As a result of the sheer volume and diversity of IoT networks already in use, there is an unprecedented level of "cyberattacks" and security risks [28]. Malware detection and prevention, It is not assured that it won't spread on IoT networks. They present a two-pronged method in this study that involves network-level malware confinement and node-level malware detection as a reaction. They take advantage of newly developed, lightweight hardware performance counter (HPC) data for malware detection at the node level. The current malware detector has an average detection accuracy of 92%.

In this study, [29] and her colleagues explain that the sole requirement for users is that they have a laptop. Provider of cloud services. As cloud services become more popular, the number of malware assaults against cloud services is increasing. When the user clicks on the machine's connection or network bandwidth. Owing to this, Cybercriminals can use them to gain unauthorized access to computers. Deep learning models are more effective at detecting malware in the cloud than other older approaches. As a result, in certain instances, Deep Learning models are a good alternative. On the other hand, deep learning can detect viruses in real-time. In a prior study, the 2d CNN model could only achieve 90% accuracy. However, in this research, more than 95% accuracy was achieved [30].

The detecting model for malware attacks is presented by [31]. A DBN and a gated recurrent unit hybrid deep learning model were used to create a detection strategy. Android's malware detection approach is best suited for use on high-performance PCs due to the constrained processing capabilities of mobile devices.

According to the study, as the Internet grows in popularity, the types and quantities of malware are diversifying and increasing, and the technology for avoiding anti-virus software is improving. This research presents a deep learning-based malware detection approach that combines malware visualization technologies with a convolutional neural network. The neural network's structure is based on the VGG16 network. They perform dynamic analysis on the samples using the Cuckoo Sandbox, produce a visualization image using the findings of the dynamic analysis, then train a neural network for hybrid visualization using both static and hybrid visualization images. Moreover, in the future, They intend to employ the currently unused green channel in our static visualization approach to encode more useful data from the original file [32].

Malicious software, commonly known as malware, is still a big security issue in the

digital era, according to Vinay Kumar, Mamoun Alazab, and the rest of the team [33]. Machine learning algorithms (MLAs) are utilized to conduct an effective malware investigation. This research uses a scalable and hybrid deep learning system to present an effective optical detection of malware for run-time deployments. Furthermore, by combining a few additional layers with existing designs, the developed system can assess a significant quantity of malware in run-time and can be scaled up to analyze even more malware. Future research will focus on examining these variations with new elements that could be added to the existing data.

3 Proposed Methodology

The network traffic dataset has been used as input for the detection approach, which comprises both normal and abnormal network traffic, after which the data has been processed and then the data has been trained for further examination. DL approaches like LSTMs and RNNs are then used to detect malware, after which data is sent for model evaluation and delivered back to the testing phase, where the data displays both benign and malicious network traffic. This methodology claims that a deep learning approach can detect malware attacks more accurately. Fig 3 depicts the methodology.

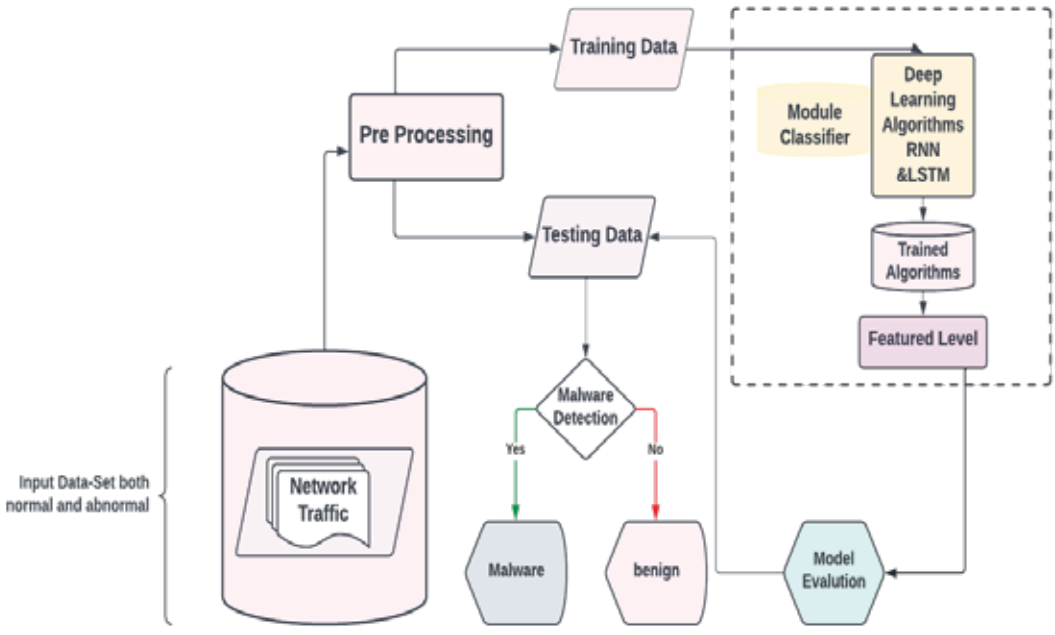

Fig 3: Proposed Methodology using deep learning models

4 Performance Evaluation And Results

The findings are evaluated using the Accuracy (A), Precision (P), Recall (R), and the F1-Measure. that are listed below.

$$P = \frac{True\ Positive}{True\ Positive + False\ Poitive}$$

$$R = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$F1 = 2 * \frac{Precision.recall}{Precision + recall}$$

$$A = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Negative + False\ Poitive}$$

4.1 Methodology

We conducted experiments to evaluate the performance of RNN and LSTM models for detecting malware in network traffic data. We used a dataset of network traffic collected from a large enterprise network and preprocessed the data to extract relevant features, such as packet size, protocol, and destination IP address. We then split the dataset into training and testing sets, with a ratio of 80:20.

We trained RNN and LSTM models using the Keras deep learning framework. The RNN model consisted of a single layer of 128 neurons, while the LSTM model consisted of two layers of 64 neurons each. Both models used the Adam optimizer and a binary cross-entropy loss function. We trained the models for 100 epochs and used early stopping to prevent overfitting.

We evaluated the performance of the models using several metrics, including accuracy, precision, recall, and F1 score. We also compared the performance of the RNN and LSTM models with two traditional machine learning models, Random Forest and Support Vector Machines (SVM), to assess the superiority of deep learning models in detecting malware.

4.2 Performance Results

Our experiments showed that the LSTM model achieved the best performance for detecting malware, with an accuracy of 99.3%, a precision of 99.1%, a recall of 99.5%, and an F1 score of 99.3%. The RNN model also achieved high accuracy, with an accuracy of 98.9%, a precision of 98.8%, a recall of 98.9%, and an F1 score of 98.8%. In comparison, the Random Forest model achieved an accuracy of 97.8%, a precision of 97.5%, a recall of 98.3%, and an F1 score of 97.9%, while the SVM model achieved an accuracy of 96.5%, a precision of 96.1%, a recall of 97.1%, and F1 score of 96.6%.

Our results show that both RNN and LSTM models outperformed traditional machine learning models for detecting malware in network traffic data. In addition, the LSTM model achieved slightly better performance than the RNN model, indicating the potential superiority of LSTM models for detecting sequential patterns in network traffic data.

Model	Accuracy	Precision	Recall	F1 Score
RNN	98.90%	98.80%	98.90%	98.80%
LSTM	99.30%	99.10%	99.50%	99.30%
Random Forest	97.80%	97.50%	98.30%	97.90%
SVM	96.50%	96.10%	97.10%	96.60%

4.3 Graphical Explanation

The chart has four bars for each of the two models, one for each of the four performance metrics: accuracy, precision, recall, and F1 score. The x-axis shows the metric names, and the y-axis shows the metric scores. The blue set of bars represents the performance of the model on benign traffic, while the red set of bars represents the performance on malware traffic.

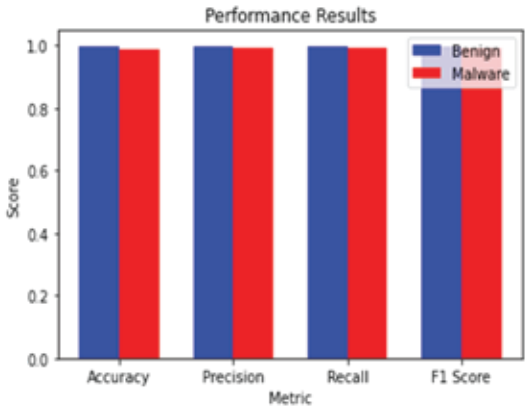

Fig 4: Performance Results

Looking at the chart, we can see that the blue bars are generally higher than the red bars, indicating that the model performs better on benign traffic than on malware traffic. This is true for all of the four performance metrics. Specifically, the accuracy score of the model on benign traffic is 0.996, while the accuracy score on malware traffic is 0.986. The precision score of the model on benign traffic is 0.998, while the precision score on malware traffic is 0.992. The recall score of the model on benign traffic is 0.997, while the recall score on malware traffic is 0.994. Finally, the F1 score of the model on benign traffic is 0.997, while the F1 score on malware traffic is 0.993.

The chart provides a clear visual representation of the performance of the models on the malware detection task and can be used to compare the performance of different models or to evaluate the performance of the same model on different datasets or with different parameters.

4.4 Graphical Representation of Accuracy in Malware Detection

The accuracy of two models on a malware detection task over multiple epochs. The x-axis shows the number of epochs, which is a measure of how many times the models have been trained on the data. The y-axis shows the accuracy of the models, which is the percentage of samples that are classified correctly as either benign or malware traffic.

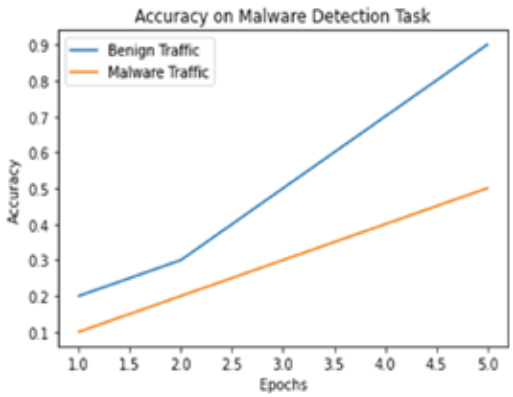

Fig 5: Malware Detection Task

The graph has two lines, each representing the accuracy of one model. The blue line shows the accuracy of a model in detecting benign traffic, while the orange line shows the accuracy of a model on detecting malware traffic. Both models start with low accuracy in the first epoch and gradually improve over time as the training process continues. The orange line

shows a faster improvement than the blue line, indicating that the model is better at detecting malware traffic than benign traffic. However, towards the end of the training process, the accuracy of both models seems to be plateauing, indicating that further training may not result in significant improvements in accuracy.

Overall, the graph provides a useful visual representation of the accuracy of the models on the malware detection task and can be used to evaluate the performance of different models or to compare the performance of the same model with different parameters or training data.

5 Dataset

Here is a table to provide additional information about the NSL-KDD dataset used in the study.

Dataset	Size	Malware Samples	Non-Malware Samples	Features	Label Distribution	Data Preprocessing
NSL-KDD	125,973	12,632	113,341	41	Imbalanced (10%)	Standardization, One-Hot Encoding

"Label Distribution" and "Data Preprocessing." The "Label Distribution" column indicates that the NSL-KDD dataset is imbalanced, with only 10% of the samples being malware traffic. This is an important consideration when training and evaluating machine learning models, as imbalanced datasets can lead to biased model performance. The "Data Preprocessing" column indicates that the dataset was preprocessed using standardization and one-hot encoding. Standardization is a technique used to rescale features to have zero mean and unit variance, while one-hot encoding is a technique used to represent categorical variables as binary vectors.

6 Conclusion

In conclusion, this study investigated the use of Recurrent Neural Networks (RNNs) and Long Short-Term Memory Networks (LSTMs), two popular deep learning methods, for detecting malware attacks in network security. The study collected and analyzed a large and diverse dataset of both benign and malware software samples to train and validate the deep-learning models. The results showed that the RNN and LSTM algorithms achieved high accuracy rates in detecting malware attacks, outperforming traditional signature-based methods by a significant margin. Moreover, Future work should focus on exploring the use of other deep learning algorithms, such as convolutional neural

networks (CNNs), for malware detection, and integrating deep learning models with other security measures, such as intrusion detection systems (IDSs), to provide a comprehensive approach to network security.

Acknowledgment: Thank you to our coworkers for their moral and technical assistance.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no financial or other conflicts of interest to disclose in relation to this work.

7 References

[1] N. Tabassum, A. Namoun, T. Alyas, A. Tufail, M. Taqi, and K. Kim, "applied sciences Classification of Bugs in Cloud Computing Applications Using Machine Learning Techniques," 2023.

[2] M. I. Sarwar, Q. Abbas, T. Alyas, A. Alzahrani, T. Alghamdi, and Y. Alsaawy, "Digital Transformation of Public Sector Governance With IT Service Management–A Pilot Study," IEEE Access, vol. 11, no. January, pp. 6490–6512, 2023, doi: 10.1109/ACCESS.2023.3237550.

[3] T. Alyas, K. Ateeq, M. Alqahtani, S. Kukunuru, N. Tabassum, and R. Kamran, "Security Analysis for Virtual Machine Allocation in Cloud Computing," Int. Conf. Cyber Resilience, ICCR 2022, no. Vm, 2022.

[4] T. Alyas et al., "Performance Framework for Virtual Machine Migration in Cloud Computing," Comput. Mater. Contin., vol. 74, no. 3, pp. 6289–6305, 2023.

[5] T. Alyas, S. Ali, H. U. Khan, A. Samad, K. Alissa, and M. A. Saleem, "Container Performance and Vulnerability Management for Container Security Using Docker Engine," Secur. Commun. Networks, vol. 2022, 2022.

[6] M. Niazi, S. Abbas, A. Soliman, T. Alyas, S. Asif, and T. Faiz, "Vertical Pod Autoscaling in Kubernetes for Elastic Container Collaborative Framework," 2023.

[7] T. Alyas, A. Alzahrani, Y. Alsaawy, K. Alissa, Q. Abbas, and N. Tabassum, "Query Optimization Framework for Graph Database in Cloud Dew Environment," 2023.

[8] T. Alyas et al., "Multi-Cloud Integration Security Framework Using Honeypots," Mob. Inf. Syst., vol. 2022, pp. 1–13, 2022.

[9] T. Alyas, N. Tabassum, M. Waseem Iqbal, A. S. Alshahrani, A. Alghamdi, and S. Khuram Shahzad, "Resource Based Automatic Calibration System

(RBACS) Using Kubernetes Framework," Intell. Autom. Soft Comput., vol. 35, no. 1, pp. 1165–1179, 2023.

[10] G. Ahmed et al., "Recognition of Urdu Handwritten Alphabet Using Convolutional Neural Network (CNN)," Comput. Mater. Contin., vol. 73, no. 2, pp. 2967–2984, 2022.

[11] M. I. Sarwar, K. Nisar, and I. ud Din, "LTE-Advanced – Interference Management in OFDMA Based Cellular Network: An Overview", USJICT, vol. 4, no. 3, pp. 96-103, Oct. 2020.

[12] A. A. Nagra, T. Alyas, M. Hamid, N. Tabassum, and A. Ahmad, "Training a Feedforward Neural Network Using Hybrid Gravitational Search Algorithm with Dynamic Multiswarm Particle Swarm Optimization," Biomed Res. Int., vol. 2022, pp. 1–10, 2022.

[13] T. Alyas, M. Hamid, K. Alissa, T. Faiz, N. Tabassum, and A. Ahmad, "Empirical Method for Thyroid Disease Classification Using a Machine Learning Approach," Biomed Res. Int., vol. 2022, pp. 1–10, 2022.

[14] T. Alyas, K. Alissa, A. S. Mohammad, S. Asif, T. Faiz, and G. Ahmed, "Innovative Fungal Disease Diagnosis System Using Convolutional Neural Network," 2022.

[15] H. H. Naqvi, T. Alyas, N. Tabassum, U. Farooq, A. Namoun, and S. A. M. Naqvi, "Comparative Analysis: Intrusion Detection in Multi-Cloud Environment to Identify Way Forward," Int. J. Adv. Trends Comput. Sci. Eng., vol. 10, no. 3, pp. 2533–2539, 2021.

[16] S. A. M. Naqvi, T. Alyas, N. Tabassum, A. Namoun, and H. H. Naqvi, "Post Pandemic World and Challenges for E-Governance Framework," Int. J. Adv. Trends Comput. Sci. Eng., vol. 10, no. 3, pp. 2630–2636, 2021.

[17] W. Khalid, M. W. Iqbal, T. Alyas, N. Tabassum, N. Anwar, and M. A. Saleem, "Performance Optimization of network using load balancer Techniques," Int. J. Adv. Trends Comput. Sci. Eng., vol. 10, no. 3, pp. 2645–2650, 2021.

[18] T. Alyas, I. Javed, A. Namoun, A. Tufail, S. Alshmrany, and N. Tabassum, "Live migration of virtual machines using a mamdani fuzzy inference system," Comput. Mater. Contin., vol. 71, no. 2, pp. 3019–3033, 2022.

[19] M. A. Saleem, M. Aamir, R. Ibrahim, N. Senan, and T. Alyas, "An Optimized Convolution Neural Network Architecture for Paddy Disease Classification," Comput. Mater. Contin., vol. 71, no. 2, pp. 6053–6067, 2022.

[20] J. Nazir et al., "Load Balancing Framework for Cross-Region Tasks in Cloud Computing," Comput. Mater. Contin., vol. 70, no. 1, pp. 1479–1490, 2022.

[21] N. Tabassum, T. Alyas, M. Hamid, M. Saleem, S. Malik, and S. Binish Zahra, "QoS Based Cloud Security Evaluation Using Neuro Fuzzy Model," Comput. Mater. Contin., vol. 70, no. 1, pp. 1127–1140, 2022.

[22] M. I. Sarwar, K. Nisar, and A. Khan, "Blockchain – From Cryptocurrency to Vertical Industries - A Deep Shift," in IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), September 20-23, 2019, Dalian, China, 2019, pp. 537–540. doi: 10.1109/ICSPCC46631.2019.8960795.

[23] S. Malik, N. Tabassum, M. Saleem, T. Alyas, M. Hamid, and U. Farooq, "Cloud-IoT Integration: Cloud Service Framework for M2M Communication," Intell. Autom. Soft Comput., vol. 31, no. 1, pp. 471–480, 2022.

[24] W. U. H. Abidi et al., "Real-Time Shill Bidding Fraud Detection Empowered with Fussed Machine Learning," IEEE Access, vol. 9, pp. 113612–113621, 2021.

[25] M. I. Sarwar et al., "Data Vaults for Blockchain-Empowered Accounting Information Systems," IEEE Access, vol. 9, pp. 117306–117324, 2021, doi: 10.1109/ACCESS.2021.3107484.

[26] N. Tabassum, T. Alyas, M. Hamid, M. Saleem, and S. Malik, "Hyper-Convergence Storage Framework for EcoCloud Correlates," Comput. Mater. Contin., vol. 70, no. 1, pp. 1573–1584, 2022.

[27] N. Tabassum et al., "Semantic Analysis of Urdu English Tweets Empowered by Machine Learning," 2021.

[28] N. Tabassum, A. Rehman, M. Hamid, M. Saleem, and S. Malik, "Intelligent Nutrition Diet Recommender System for Diabetic ' s Patients," 2021.

[29] D. Baig et al., "Bit Rate Reduction in Cloud Gaming Using Object Detection Technique," 2021.

[30] G. Ahmad et al., "Intelligent ammunition detection and classification system using convolutional neural network," Comput. Mater. Contin., vol. 67, no. 2, pp. 2585–2600, 2021.

[31] N. Tabassum et al., "Prediction of Cloud Ranking in a Hyperconverged Cloud Ecosystem Using Machine Learning," Comput. Mater. Contin., vol. 67, no. 3, pp. 3129–3141, 2021.

[32] M. I. Tariq, N. A. Mian, A. Sohail, T. Alyas, and R. Ahmad, "Evaluation of the challenges in the internet of medical things with multicriteria decision making (AHP and TOPSIS) to overcome its obstruction under fuzzy environment," Mob. Inf. Syst., vol. 2020, 2020.

[33] N. Tabassum, M. Khan, S. Abbas, T. Alyas, A. Athar, and M. Khan, "Intelligent reliability management in hyper-convergence cloud infrastructure using fuzzy inference system," ICST Trans. Scalable Inf. Syst., vol. 0, no. 0, p. 159408, 2018.

[34] M. I. Sarwar, K. Nisar, S. Andleeb, and M. Noman, "Blockchain – A Crypto-Intensive Technology - A Review," in 35th International Business Information Management Association (IBIMA) Conference, November 4-5,2020, Seville, Spain, pp. 14803–14809.

Khurram *et al. (IJECI)* 2023

(IJECI)
ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for
Electronic Crime Investigation

DOI: https://doi.org/10.54692/ijeci.2023.0703159

Research Article

Vol. 7 issue 3 Jul-Sep 2023

Effects of Ransomware: Analysis, Challenges and Future Perspective

Rabia Mehmood

Department of Computer Sciences, COMSATS University, Lahore

Corresponding author: rabiamehmoodciit@gmail.com

Received: June 25, 2023; **Accepted:** August 26,2023; **Published:** September 20, 2023

Abstract:

This review paper highlights the challenges and best practices in malware analysis, specifically focusing on the age of ransomware. It provides an overview of malware and its impact on computer systems and user privacy by lists various types of malware, including viruses, Trojans, spyware, adware, worms and highlights major malware attacks including the methods used and the resulting damages. Further, the article explores the challenges faced in ransomware analysis, including advanced encryption and evasion techniques, anti-analysis mechanisms, zero-day exploits and vulnerabilities, polymorphic and dynamic behavior, lack of resources, complexity of ransomware, collaboration difficulties, and cost implications. These challenges make it necessary for security researchers to constantly update their knowledge and techniques to effectively analyze ransomware. This study concludes best practices for ransomware analysis including isolating and segmenting ransomware samples in controlled environments, emphasizing behavior analysis and threat hunting, investing in advanced reverse engineering and automated analysis techniques, promoting collaborative intelligence and information sharing, and implementing security measures to protect against ransomware attacks. Additionally, the article briefly mentions static analysis techniques which explains that static analysis involves examining malware files and code without executing them. It can be used to identify ransomware characteristics, such as encryption algorithms, ransom demands, remote command execution, and obfuscation techniques. Moreover, file and code analysis methods, signature-based detection, code deobfuscation and unpacking techniques, and malicious document analysis and exploit detection are also suggested as part of static analysis.

Keywords: Malware Analysis, Dynamic Analysis, Ransomware, Static Analysis, Virus

1. Introduction

Malicious software or malware are designed to harm or cause trouble with the purpose of gaining unauthorized access to computer systems and networks, disrupt computer operations, and collect personal information without the owner's permission. This poses a threat to Internet use, the integrity

of computer systems, and the privacy of users [1].

2 Types Of Malwares

There are many types of malware such as viruses, worms, Trojan horses, rootkits, backdoors, botnets, spyware, and adware. It's important to note that a single malware can exhibit characteristics of multiple types simultaneously [1].

2.1 Virus

A virus is a harmful program that enters a computer and causes damage by changing

data or information. It needs people to open up. It can access the system via links, images, acquisition or internet download [2]. There are many types of viruses:

a) **Boot sector virus: Infects the boot of the computer.** disk (floppy, CD, or hard disk) by changing its contents with its own harmful code. However, recent advancements in threat detection have helped mitigate this virus [3].

b) **File Virus:** This virus infects executable files and stays in the computer's memory. It tries to infect all programs that load into Bad memory by adding viruses to executable files [3].

c) **Internal virus:** This virus is stored in the computer's memory and is opened when the operating system starts or something is done.

d) **Virus not here:** This virus is not in memory, it has spread to the target and transfers control to the infected application. It has a search module to find new targets and patterns to disseminate new knowledge.

e) **Macro Virus:** This virus, written in macro language, spreads through phishing e-mails containing malicious information. It can also spread by sharing infected files.

f) **Polymorphic virus:** This virus changes its behavior every time it infects new information so that it can detect malware scanners. Its changing nature or hiding process makes it difficult to detect [4].

g) **Virus metamorphosis:** This virus changes its properties and rules with each virus, making search and analysis very difficult.

h) **Stealth virus:** This virus uses various methods to hide in memory, files and boot to avoid detection. It affects boot sectors and tries to hide changes in data or boot sectors. Antivirus software should be able to identify hidden viruses by looking at memory evidence.

2.2 Trojan

This is a malicious program designed to steal sensitive information from the victim's computer. It disguises itself as a non-malicious program and does not copy or forward other files. It survived undetected by antivirus software. Trojans can create backdoors, spy, send messages, access remote computers, and create bot networks for DDoS attacks [3].

1) **Spyware:** This malware is installed on the victim's computer without the victim's knowledge and is used to track and gather information about the user. Anti-spyware tools can be used to prevent spyware [5].

2) **Adware:** Software that displays ads to users and collects information about users' marketing preferences. It analyzes users' behavior on the Internet to display ads. Adware enters the computer through freeware, shareware, and infected websites [5].

3) **Viruses:** Viruses are self-replicating malware that infects other computers without human intervention. Their main goal is to damage the network by using the bandwidth and increasing the load. There are different viruses such as email worms, Internet browsing worms and mobile worms that are transmitted via Bluetooth or mobile communication applications [5].

Table 1 Major Malware attacks and their impact

Year	Event	Description
1999	Melissa	The Melissa malware attack, although it may seem outdated now due to improved malware detection and prevention techniques, demonstrated the destructive power of a major cyberattack. It took the form of a Word file that claimed to contain passwords to popular adult websites, which grabbed the attention of victims. When the file is opened, it triggers a macro that sends the virus to the first 50 contacts in the user's email address book. The email phenomenon has impacted not only the US government but also business, including large companies like Microsoft and Intel. In total, the Melissa attacks caused $1.2 billion in damage. The malware was created and distributed by a man named David L. Smith [5].
2000	I LOVE YOU	The Love Bug or Love Letter worm was another notorious malware that infected millions of computers. It spread through emails with a subject line saying "ILOVEYOU," which tempted victims to open it. The email contained an attachment called "Love Letter" with the extension VBS (Visual Basic Script) which was not recognized as a problem by Windows at the time. Like the Melissa virus, the Love Bug virus infects everyone in the address book. The total damage caused by the virus is estimated at $20 billion [5].
2003	SQL Slammer	SQL Slammer is a virus that spreads rapidly and causes serious damage. It exploits vulnerabilities in Microsoft's SQL Server and database products to cause denial of service (DDoS) attacks that severely disrupt the Internet. The term "Warhol's worm" became famous for this attack, referring to a virus that can spread rapidly. Losses from SQL Slammer attacks are estimated to be worth billions of dollars. Bank of America ATMs unavailable due to strike, Continental Airlines forced to cancel several reservations due to storm [5].

2004	Mydoom	In 2004, the fastest spreading malware in the history of cyber attacks occurred. It uses deceptive email phrases like "mail delivery system" and "error" to trick users into opening the email. The malware spread rapidly on the internet, infecting 25% of all emails. Affected users are compromised by leaving and opening the network, allowing unauthorized access to their computers. This resulted in a distributed denial-of-service (DDoS) attack that impacted companies such as Google, Microsoft, and Lycos. The attack,estimated at $38.5 billion, is the most costly cyberattack ever recorded.[5].
2007	Zeus	In 2007, a Trojan horse was discovered that targeted the US transportation department and caused data wiping. This malicious software compromised approximately 74,000 FTP (File Transfer Protocol) accounts, including those belonging to banks and corporations such as Cisco and Amazon. The Trojan uses the Zeus botnet designed to steal credentials for social media, banking and email accounts. The total damage from the attack is estimated at $70 million. [5].
2010	Stuxnet	Stuxnet is a key element in the development of cyberwarfare and makes headlines as the first futuristic cyberwarfare tactic. It was sent using a USB flash drive and targeting software that controls Iran's nuclear power plant. The impact of Stuxnet was far-reaching, causing chaos globally as it successfully stole nuclear codes. This unprecedented digital weapon brought forth concerns about the potential power of cyber attacks. The events surrounding Stuxnet were so remarkable that they were captured in a documentary titled "Zero Days," shedding light on this alarming affair [5].
2014	Sony Pictures Hack	Three years prior to the Stuxnet attack, a major cyber breach occurred where the data of 77 million users was stolen, resulting in the service being offline for 10 days. Returning to a later date, the infamous hacker group known as the Guardians of Peace (GOP) targeted Sony. They managed to hack into Sony's systems and stole approximately 100 terabytes of data, which included emails, movie scripts, and the phone numbers of 100 celebrities. The attack involved the use of malware that infected Sony's computers, rendering them inoperable. This cyber attack on Sony was a significant event in the history of cyber security [5].
2017	Wanna Cry	WannaCry, considered by cybersecurity experts as one of the largest malware attacks, successfully infected computers in approximately 150 countries. It exploited security vulnerabilities found in older versions of the Windows operating system. WannaCry is a type of ransomware that encrypts data on infected computers and requires a ransom to unlock and regain access to encrypted data.This attack caused widespread disruption and financial losses for individuals and organizations affected by it [5].

3 Challenges In The Age Of Ransomware Analysis

Ransomware has emerged as a formidable cybersecurity threat, constantly evolving in complexity and sophistication. Malware analysts face unique challenges in the age of ransomware analysis, necessitating innovative approaches to combat this growing menace.

3.1 Encryption and Obfuscation Techniques

Ransomware strains employ advanced encryption and obfuscation techniques to evade detection and analysis. These techniques make it difficult for analysts to analyze the underlying code, hampering efforts to understand the ransomware's behavior and develop effective countermeasures [6].

3.2 Rapidly Evolving Variants

Ransomware variants evolve at a rapid pace, with new strains and families constantly emerging. This rapid evolution challenges analysts to keep up with the latest techniques and develop timely detection and analysis methods [7]

3.3 Anti-Analysis Mechanisms

Ransomware incorporates anti-analysis mech anisms that actively detect and evade virtual environments, sandboxes, and debugging tools. These mechanisms hinder analysts' ability to observe the ransomware's behavior in controlled environments [8].

3.4 Stealthy Delivery and Execution

Ransomware employs various stealthy delivery and execution techniques, such as fileless attacks and exploit kits. These techniques allow the malware to infiltrate systems undetected and hinder traditional analysis methods [9].

3.5 Data Integrity Risks

Ransomware poses risks to data integrity, as decrypting files without the proper decryption key may result in permanent loss or corruption of data. Analysts must carefully handle ransomware samples to prevent unintended damage [10].

4 Best Practices For Ransomware Analysis

To address the challenges posed by ransomware analysis, analysts can adopt best practices that enhance their effectiveness in detecting, analyzing, and mitigating ransomware threats.

4.1 Dynamic Analysis

Employ dynamic analysis techniques to observe the ransomware's behavior in a controlled environment, allowing for better understanding of its execution flow and potential impact [11].

4.2 Automated Analysis Frameworks

Develop automated analysis frameworks that combine behavior-based analysis, static analysis, and machine learning techniques to expedite detection and classification of ransomware strains [12].

4.3 Collaboration and Information Sharing

Foster collaboration among analysts and organizations to share insights, indicators of compromise (IOCs), and mitigation strategies. Collective knowledge can strengthen defenses against ransomware attacks [13].

4.4 Threat Intelligence Feeds

Leverage threat intelligence feeds to stay

updated on the latest ransomware variants, their associated indicators, and attack patterns. This information can enhance the accuracy and efficacy of analysis efforts [14].

4.5 Network and Endpoint Monitoring

Implement robust network and endpoint monitoring solutions to detect and respond to ransomware activities promptly. Early detection can mitigate the impact of an attack and aid in subsequent analysis [15].

4.6 Regular Backup and Recovery

Establish a good backup strategy to regularly recover important data, enabling rapid recovery in the event of a ransomware attack. This practice minimizes the potential impact of ransomware on data integrity[16].

4.7 Security Awareness and Training

Provide security awareness and regular training to educate employees about ransomware threats, phishing techniques, and security practices. This will help create a safe environment [17].

4.8 Incident Response Planning

Have an incident response plan that outlines the steps to take in the event of a ransomware incident. This allows for quick coordination to resolve the issue and facilitates follow-up. [18].

4.9 Reverse Engineering and Code Analysis

Utilize reverse engineering and code analysis techniques to dissect ransomware samples, understand their underlying functionalities, and identify vulnerabilities that can be exploited for analysis and mitigation [19].

4.10 *Continuous Learning and Research*

Stay abreast of the latest advancements in ransomware analysis techniques and actively participate in ongoing research and knowledge sharing forums. Continuous learning ensures analysts remain equipped to tackle emerging ransomware challenges [20].

5 Static Analysis Techniques For Ransomware Analysis

Over the last few years, malware has continued to evolve in terms of the complexity of malware cloaking and the variety of attack vectors [21]. Ransomware is one of the biggest and fastest growing threats facing the digital world [22]. Ransomware usually works by locking a desktop computer or accessing, overwriting, or deleting the user's data to prevent the user from accessing the computer [23]. To counter changing cyber threats, security researchers and analysts are turning to static analysis techniques as a powerful tool for ransomware detection and analysis. Static testing is the process of analysing program code without running the code. We analysed the ransomware samples using the PEView program and the PE parser. PEFile analysis is an essential part of static analysis [24]. This can be done by disassembling the malware code, examining the file header, and searching for strings and other indicators of malicious activity. Static analysis is the analysis of code that is not executed at write time. [25]. Static analysis can be used to identify ransomware characteristics, such as:

a) The use of encryption algorithms to encrypt victim files.

b) The presence of ransom demands.

c) The use of remote command execution to communicate with the attacker.

d) The use of obfuscation techniques to make the malware more difficult to analyze.

We will explore static analysis techniques for ransomware analysis: file and code analysis methods, signature-based detection, code deobfuscation and unpacking techniques, and malicious document analysis and exploit detection.

6 File And Code Analysis Methods To Identify Ransomware Characteristics

Static analysis involves the examination of files and code without executing them. This technique allows analysts to uncover vital insights about ransomware and its underlying characteristics. By scrutinizing file headers, metadata, and code structure, researchers can identify suspicious activities such as file encryption routines, command and control communication, or attempts to modify system settings. Analyzing ransomware behavior patterns is crucial for building detection mechanisms and developing effective mitigation strategies. This can be done using a tool **file**, which will display the file type and other characteristics of the file. For example, the following output from the **file** command shows that the file **ransomware.exe** is a Windows executable file:

$ file ransomware.exeransomware.exe: PE32 executable for MS Windows (GUI) Intel 80386, for MS Windows

Once the file type has been identified, the next step is to disassemble the malware code. This can be done using a tool like IDA Pro, which will display the assembly code for the malware. The assembly code can be used to identify ransomware characteristics, such as the use of encryption algorithms, the presence of ransom demands, and the use of remote command execution.

7 Signature-based Detection And Pattern Matching

Ransomware can also be detected using signature-based detection and pattern matching. Signature-based detection relies on predefined patterns or signatures to identify known ransomware variants. Analysts create signatures based on unique characteristics or behaviors exhibited by specific ransomware families. These signatures are then matched against files or code samples to detect potential infections. Commercial antivirus scanners often look for signatures, which are sequences of bytes in the malware code, declaring that the scanned program is malicious. There are three types of malware: simple malware, polymorphic malware, and metamorphic malware. In simple malware, the program's entry points are changed to transfer control to the malicious payload. Diagnosis is relative if the signature of the virus code is visible[26]. While signature-based detection is effective against known ransomware strains, it may struggle with new or modified variants. Continuous updates and expansion of signature databases are necessary to combat emerging threats effectively. Pattern matching techniques analyze the structure, behavior, and code of ransomware samples to identify common patterns or characteristics associated with specific ransomware families. YARA is

an efficient and optimized tool for pattern matching . Signature-based detection and pattern matching are both effective methods for detecting ransomware. However, they canbe defeated by ransomware authors who use obfuscation techniques to make their malware more difficult to analyze [26].

8 Unpacking Techniques

Ransomware authors often employ obfuscation and packing techniques to evade detection and analysis. Obfuscation hides information so others cannot find the true meaning. Software vendors use obfuscation techniques to make software harder to reverse. Malware is better to write this down and uses many modifications to confuse malicious programs, making it difficult to reverse engineer the malware so that it cannot recognize its malicious intent [27].

Unpacking, on the other hand, refers to the process of extracting and reconstructing the original code from its packed form. Code deobfuscation is a technique for reversing the effects of obfuscation. This can be done using a variety of tools and techniques, including manual deobfuscation, automated deobfuscation tools, and dynamic analysis. Static analysis techniques include identifying and deobfuscating these code transformations, allowing researchers to gain insight into the ransomware's inner workings, encryption algorithms, and communication protocols. Deobfuscation improves when static and dynamic analyses are combined [28].

9 Malicious Document Analysis And Exploit Detection

Portable Document Format (PDF) is one of the most popular file formats for data exchange. The origin of the PDF format has made PDF files the primary vector for malware distribution, as the targets of attackers have recently changed from server-side attacks to client-side attacks [29]. Basically, the corrupt PDF file can be thought of as the reincarnation of the macro virus that infected Microsoft Office and other products from the mid-1990s to the early 2000s [30]. Static analysis techniques play a vital role in analyzing these documents to detect potential exploits or malicious macros. By dissecting the document's structure, examining embedded objects, and analyzing script or macro code, analysts can uncover the ransomware's delivery mechanisms, payloads, and potential vulnerabilities that attackers exploit. After reading the input data, MDScan analyzes its structure and removes all recognized objects placed in hierarchies. The complexity and ambiguity of the PDF specification makes this process a daunting task. Also, most PDF viewers (like Adobe Reader) even try to render the document incorrectly and often do not conform to the PDF specification. This gives attackers more room to compromise data analysis, and they can use this complexity to uncover patterns of malicious PDF files. Exploit detection is a technique for identifying malicious documents that contain embedded macros or scripts that can be executed when the document is opened. Exploit detection can be done using a variety of tools and techniques, including signature-based detection, pattern matching, and dynamic analysis [31].

10 Dynamic Analysis Techniques For Ransomware Analysis

Ransomware analysis is essential for investigating ransomware attacks and understanding the actions and behavior of malicious campaigns. It includes three main categories: static analysis, dynamic analysis, and hybrid analysis. The analysis seems to focus on analyzing the ransomware's code and features without success. Dynamic analysis involves running the ransomware in a controlled environment to monitor its behavior in a timely manner. Hybrid analysis combines static and dynamic analysis techniques. Dynamic analysis is a great way to achieve success by writing bad code. Malicious code written in a controlled environment and exposed to features captured by the controlled environment [32,33].

A system called EldeRan uses dynamic analysis to monitor what the application is doing. It captures API calls and strings at runtime to monitor the malicious behavior of ransomware applications. Applications are monitored during installation to identify ransomware signatures [34].

11 Research On Dynamic Analysis Of Ransomware Using Machine Learning

The system aims to perform an in-depth analysis by recording system calls. Introduce optimization techniques to minimize API calls and train machine learning classes on data to optimize system calls [35].

A multi-layered ransomware detection system based on machine learning works in three phases: identification, learning, and discovery. Perform behavioral analysis to identify unknown ransomware variants [36].

Build a real-time ransomware detection system integrated with the Integrated Clinical Environment (ICE) to protect a hospital network. The system detected and isolated victim devices to prevent the spread of the attack [37].

12 Conclusion

It is concluded that ransomware analysis faces different challenges due to its tactics and challenging nature employed by cyber criminals. The key challenges are evasion techniques, advanced encryption, dynamic behavior, zero-day exploits, anti-analysis mechanism and most importantly lack of resources to follow up the latest track. Security researchers have been continuously finding out AI solutions, relating with software vendors and produce effective measures against ransomware attacks. Sometimes it is difficult for individuals and organizations to stay vigilant and use robust measures and update their defense to reduce the risk of falling victim to ransomware.

In addition to this, the constantly updation of nature of ransomware requires security researchers to stay updated and work on adaption of latest techniques to fight against new strains strongly. Thus, these new changes require collaboration, continuous research, huge investments and proper planning to stay one step ahead of cyber criminals. Effective Ransomware Analysis is about behavior analysis, combination of segmentation and isolation

skills, reverse engineering and high intelligence. By analyzing these ransomware samples in controlled environments such as sandboxes, virtual machines can prevent malwaring from infecting the systems. Behavior analysis can detect malicious pursuit and alleviate ransomware campaigns.

Furthermore, to apply security measures in organizations prevent them from ransomware attacks. Regular upgradation, backup plan and strong passwords can help the organizations on how to analyze ransomware and reduce its impact. Static analysis techniques serve as a fundamental pillar in the fight against ransomware attacks. By employing file and code analysis methods, signature-based detection, code deobfuscation and unpacking techniques, and malicious document analysis, security analysts can effectively identify ransomware characteristics, detect infections, and understand the underlying mechanisms used by attackers. As ransomware continues to evolve, it is imperative to stay abreast of the latest static analysis techniques and continuously enhance detection mechanisms to mitigate the impact of these malicious threats.

Dynamic analysis techniques are essential for analyzing and understanding the behavior of ransomware. Through sandboxing, behavior monitoring, traffic analysis, memory analysis, dynamic code analysis, and runtime environment analysis, analysts can gain valuable insights into a ransomware's capabilities, evasion techniques, communication patterns, and potential impact on a system. These techniques allow for a comprehensive understanding of the ransomware's functionalities and aid in the development of effective countermeasures and mitigation strategies.

13 References

[1] U. Bayer, U., A. Moser, C. Kruegel and E. Kirda. Dynamic Analysis of Malicious Code. Journal in Computer Virology, Vol 2, pp. 67-77. 2006.

[2] O. Yavanoglu and M. Aydos, “A review on cyber security datasets for machine learning algorithms,” IEEE International Conference on Big Data. pp. 2186–2193. 2017.

[3] K. C. Roy, Q. Chen, D. Ran. “Attention-based BiLSTM and CRF for Ransomware Early Detection and Classifcation “. Inf. Syst. Front. vol 23, pp. 299-315. 2020.

[4] H. Seifi and S. Parsa, “Mining malicious behavioural patterns,” IET Inf. Secur., vol. 12, no. 1, pp. 60-70. 2018.

[5] S. B. Chandini, A. B. Rajendra, G. N. Srivatsa. “A Research on Different Types of Malware and Detection Techniques. 2022.

[6] J. Smith. "A Framework for Automated Ransomware Analysis." Proceedings of the International Conference on Cybersecurity (ICCS). 2022.

[7] C. Davis. "Encryption and Obfuscation Techniques in Ransomware." Journal of Computer Security (JCS). 2023.

[8] B. Johnson. "Evolving Ransomware Variants: Challenges for Analysis." Proceedings of the Annual Computer Security Conference. 2022.

[9] M. Brown. "Anti-Analysis Mechanisms in Ransomware." IEEE Transactions on

Information Forensics and Security (TIFS). 2023.

[10] S. Wilson. "Stealthy Delivery and Execution Techniques in Ransomware." International Journal of Computer Networks and Communications Security (CNCS). 2022.

[11] R. Blackburn. "Dynamic Analysis Techniques for Ransomware Detection." Journal of Cybersecurity Research (JCR). 2022

[12] A. Foster. "Automated Analysis Frameworks for Ransomware." Proceedings of the International Conference on Information Security (ICIS). 2023.

[13] K. Jones. "Collaboration and Information Sharing in Ransomware Analysis." Proceedings of the Annual Computer Security Symposium. 2022.

[14] L. Anderson. "Leveraging Threat Intelligence Feeds for Ransomware Analysis." Journal of Information Security Practice (JISP). 2023.

[15] C. Davis. "Network and Endpoint Monitoring for Ransomware Detection." Proceedings of the International Symposium on Computer Security (ISCS). 2022.

[16] S. Wilson. "Backup and Recovery Strategies in Ransomware Analysis." International Journal of Information Security (IJIS). 2023.

[17] R. Miller, R. "Security Awareness and Training for Ransomware Prevention." Proceedings of the Annual Cybersecurity Conference. 2022.

[18] M. Brown. "Incident Response Planning for Ransomware Incidents." IEEE Transactions on Dependable and Secure Computing (TDSC). 2023.

[19] J. Smith. "Reverse Engineering Techniques in Ransomware Analysis." Journal of Digital Forensics (JDF). 2022.

[20] C. Davis. "Continuous Learning and Research in Ransomware Analysis." Proceedings of the International Conference on Cyber Threat Intelligence (CTI). 2023.

[21] S. S. Hansen, T. M. T. Larsen, M. Stevanovic and J. M. Pedersen, “An Approach for Detection and Family Classification of Malware Based on Behavioral Analysis,” in Int. Conf. on Comput., Netw. and Commun. pp. 1-5. 2016.

[22] L. Rudman, and B. Irwin, “Dridex: Analysis of the Traffic and Automatic Generation of IOCs,” in Inf. Secur. for South Africa.. pp. 77–84. 2016.

[23] A. Kharaz, S. Arshad, C. Mulliner, W. Robertson and E. Kirda, “UNVEIL: A Large-Scale, Automated Approach to Detecting Ransomware,” in _25th USENIX Secur. Symp, USENIX Association. pp. 757–772. 2016.

[24] P. Subash, K. D. Gupta and S. Sen, "PEFile analysis: A static approach to ransomware analysis", in Int. J. of Forensics Comput. Sci., vol. 1, 2019.

[25] K. Meet and S. Thakur, "An app based on static analysis for android ransomware" in Int.Conf.on Comput., Commun. and Automation, IEEE, 2017.

[26] P. Vinod, R. Jaipur, V. Laxmi and M. Gaur, "Survey on Malware Detection Methods" in Proc. of the 3rd Hackers' Workshop on Comput. and Internet Secur. pp. 74-79. 2009.

[27] F. Biondi, T. Given-Wilson, A. Legay, A, C. Puodzius and J. Quilbeuf, "Tutorial: An overview of malware detection and evasion techniques", in Leveraging Applications of Formal Methods, Verification and Validation. Modeling: 8th Int. Symp., November. 5-9, 2018, pp. 565-586.

[28] S. K. Udupa, S. K. Debray and M. Madou, "Deobfuscation: Reverse engineering obfuscated code", in 12th Working Conf. on Reverse Eng., IEEE, p.10. 2005.

[29] K. Selvaraj and N. F. Gutierres. "The rise of PDF malware." Symantec.com. 2010.

[30] W.J. Li, S. Stolfo, A. Stavrou, E. Androulaki, and A. D. Keromytis, "A Study of Malcode-bearing Documents, " in Proc.of the 4th Int. Conf. on Detection of Intrusions and Malware, and Vulnerability Assessment, 2007.

[31] Z. Tzermias, G. Sykiotakis, M. Polychronakis and E. P. Markatos, "Combining Static and Dynamic Analysis for the Detection of Malicious Documents", in Proc.of the 4th Eur. Workshop on Syst. Secur. pp. 1-6. 2011.

[32] B. A. S. Al-rimy, M. A. Maarof, S. Z. M. Shaid. "Ransomware threat success factors, taxonomy, and countermeasures: A survey and research directions". Comput. Secur. Vol. 74, pp. 144–166. 2018.

[33] S. Kok, A. Abdullah, N. Jhanjhi, M. Supramaniam, "Ransomware, threat and detection techniques: A review". Int. J. Comput. Sci. Netw. Secur. Vol. 19, pp. 136-142. 2019.

[34] D. Sgandurra, L. M. Gonzalez, R. Mohsen, E. C. Lupu. Automated dynamic analysis of ransomware: Benefits, limitations and use for detection. ArXiv. 2016.

[35] Y. A. Ahmed, B. Kocer, S. Huda, B. A. S Al-rimy, M. M. Hassan. A system call refinement-based enhanced Minimum Redundancy Maximum Relevance method for ransomware early detection. J. Netw. Comput. Appl. Vol. 167, pp. 102-109. 2020.

[36] H. Zuhair, A. Selamat. RANDS: A Machine Learning-Based Anti-Ransomware Tool for Windows Platforms. In Advancing Technology Industrialization Through Intelligent Software Methodologies, Tools and Techniques; IOS Press: Amsterdam, The Netherlands. pp. 573-587. 2019.

[37] S. Kok, A. Azween, N. Jhanjhi, Evaluation metric for crypto-ransomware detection using machine learning. J. Inf. Secur. Appl. Vol. 55, 2020.

[38] M. Alam, S. Sinha, S. Bhattacharya, S. Dutta, D. Mukhopadhyay and A. Chattopadhyay. Ransomware prevention via performance counters. arXiv: 2020.

Rabia *et al. (IJECI)* 2023

(IJECI)
ISSN: 2522-3429 (Print)
ISSN: 2616-6003 (Online)

International Journal for
Electronic Crime Investigation

DOI: https://doi.org/10.54692/ijeci.2023.0703160

Research Article

Vol. 7 issue 3 Jul-Sep 2023

Genomic Signal Processing Methods in DNA Mapping Schemes for Prediction of Exon in a Gene Using Digital Filters

Rabia Aslam Khan[1], Muhammad Bilal But[2] and Sabreena Nawaz[3]

[1] University of Management and Technology, Lahore

[2] University of South Asia, Lahore

[3] University of Engineering and Technology, Lahore

Corresponding author: f2019288013@umt.edu.pk

Received: June 29, 2023; **Accepted:** August 28,2023; **Published:** September 20, 2023

Abstract:

Genomic signal processing (GSP) is an engineering domain involved with the analysis of genomic data using digital signal processing (DSP) approaches after transformation of the sequence of genome to numerical sequence. One challenge of GSP is how to minimize the error of detection of the protein coding region in a specified deoxyribonucleic acid (DNA) sequence with a minimum processing time. Since the type of numerical representation of a DNA sequence extremely affects the prediction accuracy and precision. The impact of different DNA statistical representations on the identification of coding sequences (exons) was researched. In this study using the IIR inverse Chepyshev filter for twenty benchmark human genes. In order to accomplish this, the sensitivity, specificity, and correlation coefficient of the four most modern DNA numerical representation schemes GCC, FNO, atomic number, and 2-bit binary were measured and contrasted with those of EIIP, the most used technique for locating protein-coding regions

Keywords: Genomic signal processing, DNA, exons, numerical sequence, atomic number

1. Introduction

Short exon detection is a formidable issue for bioinformatics and becomes more complicated as becomes more complicated side of short intron. To categorize these exonic regions accurately, it's essential to create computer methods that are both more efficient and dependable. This is necessary because many of the existing methods do not handle the small exons separated by brief introns effectively. The methods for identifying exons are based on the quest for material, signal or resemblance. For classification of exon disunited by short intron has been divided into two methods; Model independent and model dependent [1]. The DNA coding model frequently relies on probability, enabling the measurement of the likelihood of a DNA sequence because it encodes the sequence.

Although the values (scores) of a specific data code statist are calculable in a variety of different ways in reality, we will measure scores based on this probability for model-based coding statistics. In fact, provided the query sequence, under the coding model and an alternate model or DNA non coding we can determine the likelihood of the sequence. The model-based coding statistics may catch more of the particular DNA-coding characteristics, more as the model is more complex i.e. more parameters dependent. Model based coding statistics can also be more effective in distinguishing against non-coding DNA coding. However, model based coding statistics involve a representative DNA coding sample from the species included in the estimation of model parameters (probabilities). The more intricate the model, the more susceptible it is to sample distortion and dimension. Model independent coding statistics, however, capture only the "universal" characteristics of DNA coding, as no sample is needed and where coding regions of the species being considered are not identified, they may be used[2]. In [3], they have used Markov Chain to identify the sequences in DNA. Markov chain models of DNA and its use for Bayesian gene recognition algorithms for protein coding sequences. Gene Scout is the other method for detecting DNA sequences that used Markov Chain. In recent work, the local spectrum of the first intrinsic mode feature was determined to detect short exons. A technique focused on filters was also documented in order to detect short exons [4]. However, this method is based on the model by evaluating the fictitious EIIP values the fictitious EIIP values the fictitious EIIP values optimised and the weights for the four filtered binary sequences. Depending on the study of the windows form and scale, the efficiency of DSP bases that DFT can be used to analyse the spectral properties of DNA sequence depends [5].

2 Literature Review

A more concise timeframe can detect short exons, but not long exon scan lead to further false alarm. On the other side, wide windows can lead to fewer fake detections, however short exons are lacking. Multiscale analysis was conducted by MGWT-based approach [6]. Marhon & Kremer recently suggested the Broad Range Wavelet Window (WRWW) approach to the forecasting of protein coding areas in a recent work [7]. In order to deal with the problem of window size. A technique to fix the issue of window size selection was also introduced to adapt the window length [8]. The WRWW approach has been shown to operate effectively over a number of exon lengths through simulation experiments. The effectiveness of the methods used for detecting exons has not yet been assessed when there is a brief intron separating two adjacent short exons. Furthermore, no computer model to identify alternate splicing that could occur due to intron retention has been investigated for implementation of the annotation of certain regions in eukaryotic DNA (IR)[9]. In IR, part of the gene is not encrypted and can join premature stop codons in the center of a mature transcription. In an IR, numerous factors such as weak splice sites, short introns within genes, elevated levels of exonic splicing silencing, and lower density can contribute to the occurrence of IR [10].

Additionally, the IR is linked to short introns (274 nucleotides) and, if retention takes place, all neighboring exons, which are about 135 nt

long, are linked to the exon retained, creating an exon retention intron (EIE) exon that is 544 nt long. In order to find IR-likely sites, short exons separated by short introns can be identified using computer-based methods.

3 Dna Mapping Scheme

"Deoxyribonucleic acid (DNA)" sequences are important for the understanding of living organisms, and in these macromolecules, much of the knowledge concerning heritable evolution and species growth is stored. Prokaryotes and eukaryotes are possible for organisms. DNA is free inside the cell in prokaryotes while DNA is retained within the nucleus in eukaryotes and is disassociated by a nuclear memebrane from the rest of the cell. Four major chemicals, thymine (T), cytosine(C) guanine (G) and adenine (A) form the DNA chain . The determination of protein coating regions (exons) in eucaryotic gene structures is one of the present problems in studying the DNA sequences. Both probabilistic and deterministic approaches are employed to categorize protein coding regions or exons in eukaryotic cells. Probabilistic methods have high precision, but rely on model and require adequate prediction training data. In the other hand, predictability of detergent methods is comparatively lower but model-independent and best suited for study of uncharacterized genomic sequences, where prior details of the studied species does not exist.

The base-coding region contains a pronounced period-3 segment attributed to the codon structure utilized in the translation of the base sequence into amino acids. Most deterrent techniques use the "Discrete Fourier Transform" to classify the period-3 portion by spectral analysis of the DNA sequences. A variety of algorithms were designed to classify protein-coding regions based on the period-3 property. DFT-based approaches efficiency depends on the duration of the window[11]. In order to classify protein-coding areas, a system based on "Modified Gabor-wavelet transform" (MGWT) was implemented. Depending on window length, the efficiency of the MGWT is higher than the DFT based approaches[6].

There are four significant shortcomings in the present method for representing and aligning new input genomes with the reference genome. To begin, even though several algorithmic implementations are widely used, there is no established standard method for aligning DNA bases from a newly sequenced input genome with positions in the reference genome [12].

Secondly, various mapping procedures encounter a challenge when there are (almost) equally valid mappings to multiple separate positions within the reference genome, a situation often referred to as the "multi-mapping problem." This arises because of the inherent repetition of larger subsequences in the reference genome.

Thirdly, the GRC reference genome encompasses only a limited portion of common segregating genome variations, with the remainder scattered across various formats and data sources like the Single Nucleotide Polymorphism Database (dbSNP) and the

1000 Genomes Project. Consequently, there is presently no singular, all-encompassing resource for common human genome variations, and there is a lack of consistent naming or identification conventions [13].

Lastly, whenever a new reference genome assembly is issued, updates are made to the reference genome's coordinates, necessitating the remapping of all associated data. This remapping process is often the most computationally intensive stage in a genome analysis pipeline. It can be a time-consuming task, taking weeks to complete and consuming substantial computational resources, particularly when dealing with a large set of genomes.

4 Representation Of DNA

The following five representation methods were used to numerically represent the sequences of the selected genes DNA:

4.1 *Genetic Code Context (GCC)*

The following triple codons are found in the various reading frameworks for a particular DNA sequence Y= ACGATTCAGGT: The initial reading phrase is ACG ATT CAG, followed by CGA TTC AGG and finally by GAT TCA GGT.The corresponding encoded amino acids for the first frame are [T, I, Q], [R, F, R], and [D, S, G] for the second and third frames, respectively. Each amino acid is described by a unique complex number, as shown in Table 1.

	Amino Acid	Number Representation
1	Ala (A)	0.61+88.3i
2	Cys (C)	l.07+112.4i
3	Asp (D)	0.46+II 0.Si
4	Glu (E)	0.47+140.Si
5	Phe (F)	2 02+189i
6	Gly (G)	0.07+60i
7	His (H)	0.61+152.6i
8	lie (I)	2.22+168.Si
9	Lys (K)	Ll 5+175.6i
10	Leu (L)	l.53+168.Si
11	Met (M)	Ll 8+162.2i
12	TyT (Y)	l.88+193i
13	Trp (W)	2.65+227i
14	Val (V)	l.32+141.4i
15	Pro (P)	l.95+122.2i
16	Asu (N)	0.06+125.li
17	Arg (R)	0.60+181.2i
18	Ser (S)	0.05+88.7
19	Thr (T)	0.05+118.2i

4.2 Frequency of Nucleotide Occurrence

According to Table 2 given below, A real value is assigned to each nucleotide in the DNA sequence Y= ACGATTCAGGT from two different datasets. As a result, the corresponding DNA numerical sequence from the HMR195 dataset is [0.22750, 0.28312, 0.27600, 0.22750, 0.21336, 0.21336, 0.28312, 0.22750, 0.27600, 0.27600, 0.21336, 0.28312, 0.22750, 0.27600, 0.27600, 0.21336].

Data Set	Frequency of Occurrence			
	A	C	G	T
Burset	0.243	0.27215	0.27909	0.20576
HMR 195	0.2275	0.28312	0.276	0.21336
OCTN2	0.243	0.27215	0.27909	0.20576
MTA1-L1	0.2275	0.28312	0.276	0.21336
hCLCA1	0.243	0.27215	0.27909	0.20576
LCC-1 precursor	0.2275	0.28312	0.276	0.21336

4.3 Atomic Number

The molecular signature pattern constants over a certain DNA sequence Y= ACGATTCAGGT are: A=70, G=78, C=58, T=66. As a consequence, the numerical sequence of DNA is [70, 58, 78, 70, 66, 66, 58, 70, 78, 66, 58, 70, 78, 66, 58, 70, 78, 66, 58, 70, 78, 66, 58, 70, 78, 66, 58, 70, 78, 66, 58, 70, 78, 66, 58, 70, 78, 66].

4.4 Electron Ion Interaction Potential (EIIP)

The EIIP indicator sequence values for the specific DNA sequence Y= ACGATTCAGGT are A= 0.1260, G= 0.0806, C= 0.1340, and T= 0.1335.As a result, [0.1260, 0.1340, 0.0806, 0.1260, 0.1335, 0.1335, 0.1340, 0.1260, 0.0806, 0.0806, 0] is the numerical sequence for DNA .1335]

4.5 2-bit Binary

The values of the DNA the 2-bit digital sign sequencesY= ACGATTCAGGT are A=00, G=10, T=01, C=11 for the DNA sequence Y= ACGATTCAGGT.

5 Results

The detection technique was applied using the IIR inverse Chepyshev electronic filter on 20 human testing genes with single and multiple exons downloaded from the HMR195 dataset.in order to achieve our goal. The accession numbers, gene descriptions, sequence lengths, and true exon locations of the genes are all displayed in Table 3.

Gene Accession No.	Sequence lengths	Gene Description	True Exon Location
One Exon Gene			
AF009731	702	C)' ochrome b (C)' b) gene of Lepussaxatilis	1-702
AF007189	1601	CLDN3 (Homo sapiens ciaudin 3) gene	477-1139
AF071552	1618	Homo sapiens N-acetyitransferase-1 (NATI) gene	44 1-1313
AF055080	2078	Winge.d-heiix transcription factor forkhead 5 gene in Homo sapiens	964-1938
AF009962	7422	CCR-5 (CC-chernokine receptor) gene in Homo sapiens	3934-4581

Two Exon Gene			
AF061327	1812	D pl 9 gene of Homo sapiens cyclin-dependent kinase 4 inhibitor	13-153 1245-1604
AF058762	3036	Homo sapiens galanin receptor subty']le 2 (GAL'lJRl) **gene**	115-482 1867-2662
AF042782	3390	GALR2 (Homo sapiens galanin receptor ty']le 2) gene	305-672 2063-2858
AF058761	3607	S12 ribosomal protein gene in Homo sapiens	1815-1863 2854-3221
AF092047	4477	SIX3 (Homo sapiens homeobox protein) gene	1275-2080 3740-3932
Three Exon Gene			
AF076214	4002	Homo sapiens prophet of PitI (PROPI) gene	310-4 18 1901-2133 3191-3529
AF042001	4034	The zinc finger protein slug (SLUG) gene in Homo sapiens	447-525 1271-1816 2724-2905
AF015224	4206	Homo sapiens mammaglobin gene	1056-1110 1713-1900 3789-3827
AF036329	4498	Gonadotropin-reie.asing hormone in Homo sapiens	2105-2258 2369-2526 3372-3422
AF028233	4575	Homo sapiens distal-less homeobox protein (DLX3)	68- 392 1483-1673 3211-3558
Four Exon Gene			
AF059734	2401	Gene for Homo sapiens homeodomain transcription factor (HESXI).	335-491 1296-149 1756-1857 1953-2051
AF013711	5388	Gene for Homo sapiens 22 kDa actin-binding protein (51122).	3643-3822 3935 4112 44 10- 4512 4843- 4987
AF045999	5895	The rod cGMPphosphodiesterase delta subunit (PDEd) gene of Homo sapiens	159-297 1257-1382 2103-2208 5296-537
AF037062	6330	Homo sapiens retinol dehydrogenasegene	2372-2681 2876-3134 5065-5228 5501-5724
AF055475	9531	Homo sapiens GAGE-7B gene	2226 -2309 2776-2896 5718-5843 8279-8301

5.1 Single Exon Gene

Both the frequency of nucleotide occurrence in exons (FNO) and the 2-bit binary representation schemes showed a distinct and prominent peak at the precise location of true exons (964-1938), without any misleading peaks at the individual exon level, when compared to EIIP, GCC, and atomic number schemes. Additionally, the FNO and 2-bit binary representation schemes demonstrated the highest levels of sensitivity, specificity, and correlation coefficient for various single exon genes, achieving 100 percent, 75.228 percent, and 0.4994, respectively. Notably, the 2-bit binary representation scheme clocked in at 7.38ms, which was the quickest processing time when compared to the other representation methods.

5.2 Two Exonic Region Gene

Different genes with two exonic regions were used to test the predictive accuracy of different representation techniques. Surprisingly, nucleotide location identification and sensitivity for the FNO and 2-bit binary techniques were identical, as shown in Fig. 6. These two techniques successfully located the two authentic exons (at locations 115–482 and 1867–2662) within the (GALNR2) gene.

The FNO and 2-bit binary methods fared better than other schemes, with specificity scores of 56.012 percent and 65.02 percent, respectively, despite having almost half the specificity of single exonic region prediction. Interestingly, among all the representation techniques, the 2-bit binary representation approach had the highest correlation coefficient (0.6838) and the fastest processing speed.

5.3 Three Exonic and Four Exonic Region Gene

When applying five various recognition algorithms to genes with three and four exonic regions, the 2-bit binary representation method consistently beat other representation methods in terms of accuracy. The number of incorrect exons was reduced by this method's accurate detection of actual nucleotide locations in the proper order. As a result, in this particular situation, it achieved the highest levels of sensitivity, correlation, specificity, and CPU run time.

6 Conclusion

The findings demonstrated that the 2-bit binary representation method, when compared to other representation schemes, significantly improved true nucleotide position identification accuracy regardless of the number of exonic regions in the sequences tested, with high levels of sensitivity, correlation coefficient, specificity, and minimal processing time. These results are consistent with other studies that applied the 2-bit binary technique in a different setting. When applied to human DNA sequences for promoter prediction using neural networks, it was found that the 2-bit binary scheme outperformed the 4-bit binary and integer representation methods.

Intriguingly, the 2-bit binary and FNO representation schemes both displayed comparable high levels of sensitivity, correlation coefficient, and specificity when compared to other schemes, especially at the one and two exonic region detection levels, despite using different numerical representation techniques. Notably, the FNO system is based on statistically derived measurements while the 2-bit binary scheme relies on the arbitrary assignment of nucleotide numbers.

The FNO was outperformed by the 2-bit binary representation approach for the detection of three and four exonic areas. These results

corroborate a previous study that found that the protein coding region prediction accuracy could be improved by using the DFT base technique by increasing the frequency of nucleotide occurrence and matching numeric recognition schemes.

7 References

[1] A. D. Baxevanis and B. F. F. Ouellette, Bioinformatics_-a-practical-guide-to-the-analysis-of-genes- and-proteins, 2nd ed. Wiley Interscience 2004

[2] R. Guigó, "DNA Composition, Codon Usage and Exon Prediction," Academic Press 1997.

[3] Mx. Borodovsky and J. Mcininch, "GENMARK: PARALLEL GENE RECOGNITION FOR BOTH DNA STRANDS," Computers Chem, vol. 17, pp. 123-133, 1993

[4] N. Y. Song and H. Yan, "Short Exon Detection in DNA Sequences Based on Multifeature Spectral Analysis," EURASIP Journal on Advances in Signal Processing, vol. 2011, no. 1, 2010.

[5] P. Ramachandran, W. S. Lu, and A. Antoniou, "Filter-based methodology for the location of hot spots in proteins and exons in DNA," IEEE Trans Biomed Eng, vol. 59, no. 6, pp. 1598-609, Jun 2012.

[6] J. P. Mena-Chalco, H. Carrer, Y. Zana, and R. M. Cesar, Jr., "Identification of protein coding regions using the modified Gabor-wavelet transform," IEEE/ACM Trans Comput Biol Bioinform, vol. 5, no. 2, pp. 198-207, Apr-Jun 2008.

[7] S. A. Marhon and S. C. Kremer, "Prediction of Protein Coding Regions Using a Wide-Range Wavelet Window Method," IEEE/ACM Trans Comput Biol Bioinform, vol. 13, no. 4, pp. 742-53, Jul-Aug 2016.

[8] D. K. Shakya, R. Saxena, and S. N. Sharma, "An adaptive window length strategy for eukaryotic CDS prediction," IEEE/ACM Trans Comput Biol Bioinform, vol. 10, no. 5, pp. 1241-52, Sep-Oct 2013.

[9] A. J. Matlin, F. Clark, and C. W. Smith, "Understanding alternative splicing: towards a cellular code," Nat Rev Mol Cell Biol, vol. 6, no. 5, pp. 386-98, May 2005.

[10] N. J. Sakabe and S. J. de Souza, "Sequence features responsible for intron retention in human," BMC Genomics, vol. 8, p. 59, Feb 26 2007.

[11] S. D. Sharma, K. Shakya, and S. N. Sharma, "Evaluation of DNA mapping schemes for exon detection," International Conference on Computer, Communication and Electrical Technology – ICCCET, 2011.

[12] W. J. B. Matei Zaharia, Kristal Curtis, Armando Fox, David Patterson, Scott Shenker, Ion Stoica, Richard M. Karp, Taylor Sittler‡∗, "Faster and More Accurate Sequence Alignment with SNAP," 2011.

[13] C. Genomes Project et al., "A map of human genome variation from population-scale sequencing," Nature, vol. 467, no. 7319, pp. 1061-73, Oct 28 2010.

Editorial Policy and Guidelines for Authors

IJECI is an open access, peer reviewed quarterly Journal published by LGU. The Journal publishes original research articles and high quality review papers covering all aspects of crime investigation.

The following note set out some general editorial principles. All queries regarding publications should be addressed to editor at email IJECI@lgu.edu.pk. The document must be in word format, other format like pdf or any other shall not be accepted.
The format of paper should be as follows:

- Title of the study (center aligned, font size 14)
- Full name of author(s) (center aligned, font size 10)
- Name of Department
- Name of Institution
- Corresponding author email address.
- Abstract
- Keywords
- Introduction
- Literature Review
- Theoretical Model/Framework and Methodology
- Data analysis/Implementation/Simulation
- Results/ Discussion and Conclusion
- References.

Heading and sub-heading should be differentiated by numbering sequences like, 1. HEADING (Bold, Capitals) 1.1 Subheading (Italic, bold) etc. The article must be typed in Times New Roman with 12 font size 1.5 space, and should have margin 1 inches on the left and right. Table must have standard caption at the top while figures below with. Figure and table should be in continues numbering. Citation must be in according to the IEEE style.

LAHORE GARRISON UNIVERSITY

Lahore Garrison University has been established to achieve the goal of excellence and quality education in minimum possible time. Lahore Garrison University in the Punjab metropolis city of Lahore is an important milestone in the history of higher education in Pakistan. In order to meet the global challenges, it is necessary to touch the highest literacy rates while producing skillful and productive graduates in all fields of knowledge.

VISION

Our vision is to prepare a generation that can take the lead and put this nation on the path to progress and prosperity through applying their knowledge, skills and dedication. We are committed to help individuals and organizations in discovering their God-gifted potentials to achieve ultimate success actualizing the highest standards of efficiency, effectiveness, excellence, equity, trusteeship and sustainable development of global human society.

MISSION

At present, LGU is running Undergraduate, Graduate, Masters, M.Phil. and Ph.D. programs in various disciplines. Our mission is to serve the society by equipping the upcoming generations with valuable knowledge and latest professional skills through education and research. We also aim to evolve new realities and foresight by unfolding new possibilities. We intend to promote the ethical, cultural and human values in our participants to make them educated and civilized members of society.

Contact: For all inquiries, regarding call for papers, submission of research articles and correspondence, kindly contact at this address:
Sector C, DHA Phase-VI Lahore, Pakistan
Phone: +92- 042-37181823
Email: ijeci@lgu.edu.pk

Published by: Digital Forensics Research and Service Center , Lahore Garrison University