 ISBN: 978-978-53006-97

AKSU JOURNAL OF ENGINEERING AND TECHNOLOGY

Faculty of Engineering
Akwa-Ibom State University
Ikot Akpaden, Mkpat Enin L.G.A
Akwa Ibom State, Nigeria.

VOL. 1 NO 1, APRIL 2026

FOREWORD

It is with great pleasure and a deep sense of responsibility that I present this edition of the *AKSU Journal of Engineering and Technology*. This journal continues to serve as a vital platform for the dissemination of high-quality research, innovative ideas, and practical solutions within the fields of engineering and technology.

In an era defined by rapid technological advancement and complex global challenges, the role of research in shaping sustainable development cannot be overstated. The contributions featured in this edition reflect rigorous scholarly efforts, addressing contemporary issues and offering insights that are both relevant and impactful. The journal remains committed to promoting interdisciplinary collaboration, encouraging intellectual curiosity, and fostering innovation among researchers, academics, and industry practitioners.

I commend the authors for their dedication and the reviewers for their thorough and objective evaluations, which have significantly contributed to maintaining the high standards of this publication. I also appreciate the efforts of the editorial board and all those who have worked tirelessly behind the scenes to ensure the successful release of this edition.

It is my sincere hope that the articles contained herein will not only advance knowledge but also inspire further research and practical applications that will benefit society at large.

Thank you for your continued support and commitment to academic excellence.

Dr. Enyenihi H. Johnson
Editor-in-Chief
AKSU Journal of Engineering and Technology

EDITORIAL BOARD MEMBERS

 ISBN: 978-978-53006-97

IMPORTANT NOTE

The Faculty of Engineering, Akwa-Ibom State University, IkotAkpaden, Nigeria does not assume responsibility for the points of view or opinions of the contributors unless such statements have been established by resolution. All correspondence regarding this issue should be sent to

Editor in-chief
AKSU Journal of Engineering and Technology,
Akwa-Ibom State University,

LIST OF CONTRIBUTORS

1. Etim, Edayeobong R, Department of Electrical and Electronic Engineering, AkwaIbom State University, IkotAkpaden, Nigeria; raphetim@gmail.com

2. Ekpe, Unwana M, Department of Electrical and Electornic Engineering, AkwaIbom State University, IkotAkpaden, Nigeria, and Department of Computer Engineering, Makerere University, Kampala, Uganda; unwanaekpe@aksu.edu.ng

3. Johnson, Enyenihi H , Department of Electrical and Electronic Engineering, AkwaIbom State University, IkotAkpaden, Nigeria; enyenihijohnson@aksu.edu.ng

4. Akpasam J. Ekanem, Department of Electrical and Electronic Engineering, AkwaIbom State University, IkotAkpaden, Nigeria; akpasamekanem@aksu.edu.ng

5. Okon Nsa Ufot, Department of Computer Engineering Technology, AkwaIbom State Polytechnic, Ikotosurua, ufot.okon@akwaibompoly.edu.ng.

TABLE OF CONTENTS

Satellite-Enabled 6G Networks: Bridging the Digital Divide in Hard-to-Reach Areas of Nigeria

Etim, Edayeobong R.*[1], Ekpe, Unwana M[2], and Johnson, Enyenihi H.[3]
[1]Department of Electrical and Electronic Engineering, Akwalbom State University, IkotAkpaden, Nigeria; raphetim@gmail.com
[2]Department of Electrical and Electornic Engineering, Akwalbom State University, IkotAkpaden, Nigeria, and Department of Computer Engineering, Makerere University, Kampala, Uganda; unwanaekpe@aksu.edu.ng
[3]Department of Electrical and Electronic Engineering, Akwalbom State University, IkotAkpaden, Nigeria; enyenihijohnson@aksu.edu.ng
*Corresponding author's email: ralphetim@gmail.com. +2348081133169

ABSTRACT
In the face of Nigeria's expeditious urban digital growth, many rural and remote communities remain isolated from communication services. This paper analyzes how satellite-enabled sixth generation (6G) networks can revolutionize connectivity in Nigeria's hard-to-reach areas.These are rural areas and settlements in the Niger Delta, the Chad Basin, and mountainous areas bordering the Republic of Cameroun, where terrestrial cellular and cable-based networks do not currently provide service due to difficulties in installing ground-based telecommunications infrastructure. The paper examines the current satellite initiatives, especially how Geostationary (GEO) and Low Earth Orbit (LEO) satellite constellations can complement terrestrial networks using enabling technologies and policy frameworks to offer a roadmap for all-inclusive connectivity. This study employs a quantitative engineering approach to evaluate the role of satellite communication systems in extending mobile connectivity to hard-to-reach areas in Nigeria, with a focus on 6G integration. The key parameters to ensure ubiquitous 6G coverage are embedded in the link budget analysis. Using MATLAB and Excel-based modeling, satellite visibility, slant path geometry, and achievable throughput for GEO and LEO systems are examined. Four satellites (NigComSat-1R, Echostar 16, Astra 1A, and Echostar 6) were confirmed to be visible from the Niger Delta region, with Uyo, taken as a representative city. The elevation angles of the satellites ranged from 38.0° to 46.5° and the slant path distances were between 40,987.12 km and 41,550.88 km. NigComSat-1R's C-band service yielded downlink/uplink C/N ratios of 20.36 dB and 24.61 dB respectively, enabling Internet of Things (IoT) data rates of 134–266 kbps (DL) and 6.0–10.05 kbps (UL). In contrast, LEO-based Ku-band OneWeb services achieved 140–880 Mbps, validating their potential for community backhaul. The study affirms satellite integration as a technically viable solution for ubiquitous mobile coverage in 6G networks.

KEYWORDS: *6G, GEO, LEO, IoT, Non-terrestrial Networks, Satellite Communication.*

1. INTRODUCTION

Africa's most populous nation, Nigeria, faces a serious digital divide. While the urban centers enjoy expanding broadband access, rural and hard-to-reach areas struggle with limited and poor connectivity , . The increasing demand for high-speed, reliable, and ubiquitous mobile communication has exposed the limitations of traditional terrestrial infrastructure, especially in regions with difficult terrain and low population density, where it may be economically unfeasible to deploy ground-based stations . The emerging 6G framework emphasizes the inclusion of non-terrestrial networks (NTNs) to complement terrestrial infrastructure, and satellite-enabled platforms can meet this demand as they have been shown to achieve very wide coverage, relatively low latency, and appreciable throughput , . This paper presents an analysis of satellite-based systems for enhancing connectivity in Nigeria's underserved areas, reflecting the need to urgently close the digital divide in the country.

2. THEORETICAL BACKGROUND AND REVIEW OF RELATED WORKS

Nigeria's digital divide is created by geography, income and infrastructure , . According to the Nigerian Communications Commission (NCC), broadband penetration stood at 48.81% as of mid-2025, with rural areas lagging significantly behind . The following sections provide an overview of the evolution of mobile communication systems, discusses how satellite networks are being integrated into terrestrial networks, and reviews some enabling technologies that make the integration possible before the review of some related works.

2.1. MOBILE COMMUNICATIONS: FROM 1G TO 6G

The evolution of mobile technology spans six generations. From the analogue first generation (1G) systems of the 1970s, each generation has introduced groundbreaking features. For example, second generation (2G) systems was anchored by the Global System for Mobile (GSM) communications technology. It ushered in digital services such as Short Message Services (SMS) and Multimedia Services (MMS). The third generation (3G) brought about mobile broadband capabilities based on the Wideband Code Division Multiple Access (WCDMA) and High Speed Packet Access (HSPA) technologies. The main fourth generation (4G) technology was the Long Term Evolution (LTE) and its advanced iteration, LTE-A. These technologies have significantly improved the throughput and latency capabilities of 3G, and the fifth generation (5G) promises even higher capabilities in terms of enhanced mobile broadband (eMBB), ultra-reliable and low-latency communications (URLLC), and massive machine-type communications (mMTC) . The sixth generation (6G) is expected to build on present capabilities by integrating satellite and terrestrial systems for ubiquitous coverage.

2.2. SATELLITE COMMUNICATIONS NETWORKS IN NIGERIA

Satellites in Low Earth Orbit (LEO), Medium Earth Orbit (MEO), and Geostationary Earth Orbit (GEO) facilitate navigation, weather forecasting, and global communication. Nigeria's major satellite communication infrastructure is the NigComSat-1R, which is a GEO satellite with C-band and Ku-band transponders. A

recent partnership between Eutelsat and the Nigerian Communications Satellite (NigComSat) Ltd aims to leverage on the OneWeb LEO constellation to deliver high-speed low-latency connectivity across the country .

The main frequency bands used in satellite communications are shown in Table 1, while Figure 1 shows a typical non-terrestrial communication network with space, air, and ground components . The space-borne components are satellites in GEO and non-geosynchronous orbits (NGSO), while the air-borne platforms can be unmanned aerial vehicles (UAVs) or high altitude platforms (HAPs). The satellites can either be of the regenerative type, whereby digital signal process is carried out onboard the satellite, or of the bent-pipe (transparent) architecture. Both architectures are capable of supporting the next generation radio access network (NG-RAN).

TABLE 1: FREQUENCY BANDS FOR SATELLITE COMMUNICATIONS

Frequency Band	Downlink (GHz)	Uplink (GHz)
L	1.53–1.559	1.6265–1.6605
S	2.50–2.655	2.655–2.69
C	3.40–4.20	4.50–4.80; 5.725–7.075
X	7.25–7.75	7.90–8.40
Ku	10.70–12.75	12.75–13.25; 14.0–14.8
K	18.1–21.2	17.3–18.1
Ka/Q/V	37.5–40.5	27.0–31.0; 42.5–43.5; 47.2–50.2; 50.4–51.5

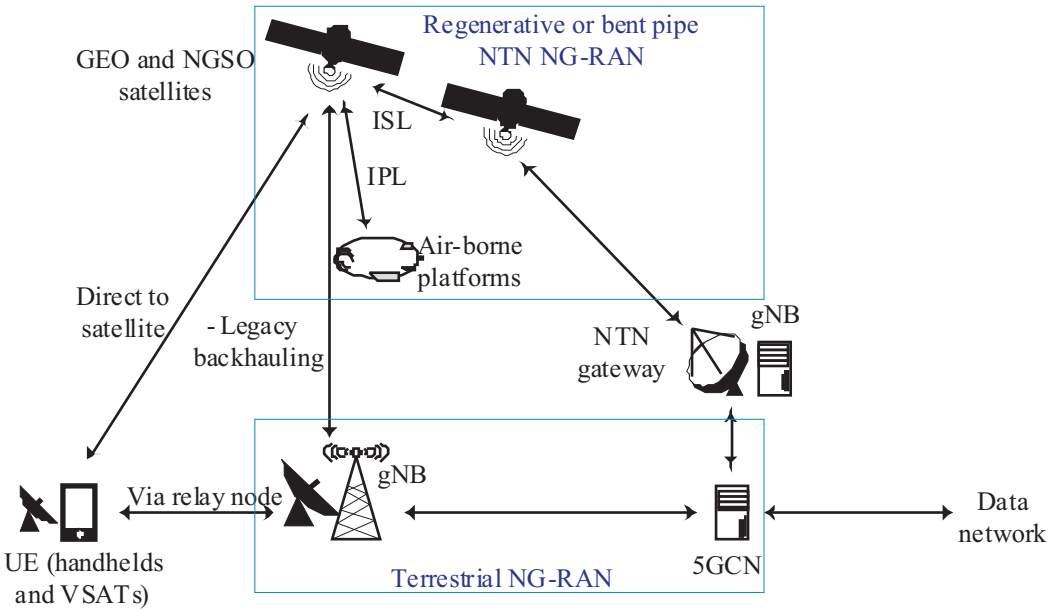


Figure 1: A non-terrestrial network showing space, air, and ground components

2.3. INTEGRATION OF MOBILE AND SATELLITE SYSTEMS FOR 6G COMMUNICATIONS

The integration of non-terrestrial satellite and mobile terrestrial networks is central to 6G's vision and this is expected to deliver ultra-fast speeds, low latency and ubiquitous connectivity.

The ensuing hybrid networks would employ GEO satellites for wide coverage and LEO constellations for low latency while artificial intelligence (AI), edge computing, reconfigurable intelligent surfaces (RIS), and other enabling technologies would be leveraged to enable the networks to reach areas beyond traditional terrestrial network coverage . This will enable applications such as autonomous vehicles, holographic communications, and high-fidelity virtual reality . Figure 2 depicts a multilayered integrated satellite-aerial-terrestrial RIS-empowered communication system .

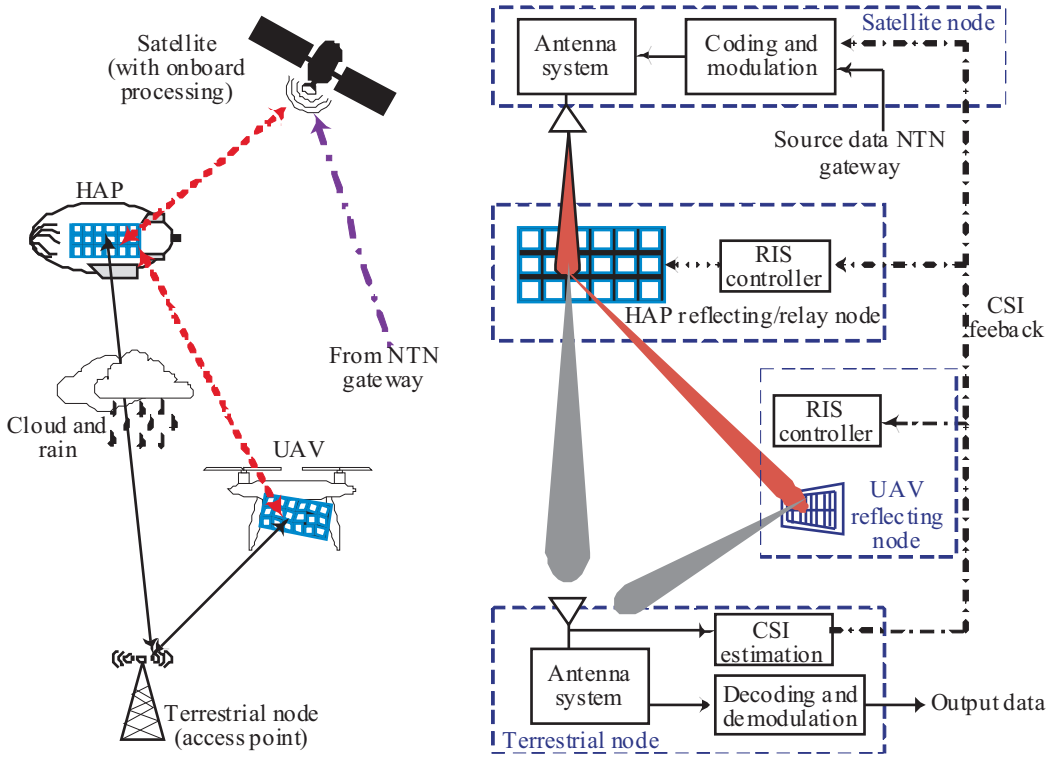


Figure 2: Integrated satellite-aerial-terrestrial RIS-empowered communication system pictorial and block diagram

2.4 HARD-TO-REACH AREAS

Over 40% of the global population resides in rural or remote areas with limited or poor

mobile coverage . In Nigeria, some of such remote locations fall under the category of hard-to-reach areas, and examples can be found in the Niger Delta, the Chad Basin, and the mountainous areas bordering the Republic of Cameroun. These locations are mainly characterized by difficult geography and challenging terrain. In addition, some of the hard-to-reach areas are plagued with security concerns due to banditry and insurgency , high telecommunications infrastructure deployment costs, and a projected low returns on investments due to sparse population and low purchasing power of the locals. Where terrestrial mobile communications connectivity exists, service has been reported to be intermittent, unreliable, and unsatisfactory .

Fortunately, satellite networks can bridge the connectivity gap in hard-to-reach areas but face challenges such as high deployment cost of satellite infrastructure alongside high latency and spectrum congestion. This is especially the case for GEO-based systems. However, a viable solution to bridging the coverage gap is by integrating satellites into existing terrestrial networks , . Some of the key enabling technologies for the convergence of terrestrial and non-terrestrial networks include Software Defined Radio (SDR) for flexible signal processing , Cognitive Radio (CR) for intelligent spectrum utilization, and Rate-Splitting Multiple Access (RSMA) for interference management and multi-user optimization . Other important technologies include Network Function Virtualization (NFV) for cost-efficient deployment , Reconfigurable Intelligent Surfaces (RIS) , and massive MIMO for enhanced connectivity.

2.5. RELATED WORKS AND RESEARCH GAPS

This section groups research works that have attempted to solve the hard-to-reach problem into interference management attempts, standardization, enabling technologies, coverage extension, mobility management, and security.

In terms of interference management, it has been identified that integrating satellites into terrestrial communication networks can solve the hard to reach problem. One of the research works that examined this solution is reported in . This paper presented the emergence of RSMA as a powerful multiple access, interference management and multi-user strategy for next generation integrated satellite-terrestrial communication systems. It showed how RSMA can be used to manage interference in a multi-user 6G communications system. It contrasted the numerous benefits RSMA offers against the past generations of multiple access techniques.

Standardization being one of the main enablers for the integration of satellites and terrestrial networks has been extensively discussed in . The paper presents an overview of the role that LEO satellite mega constellations would play in providing ubiquitous internet and communication services in the future. The paper reviews the 3GPP standardization activities for integrating such constellations into the 5G network and gives possible standardization directions for future 6G systems. It also reviews the standardization efforts of organizations other than the 3GPP, and went further to show how LEO satellite networks will provide wide-area coverage and support service availability, continuity and scalability.

There are several enabling technologies that can help address the challenges of hard-to-reach areas. According to , solutions can be provided in terms of increasing service agility, and reducing the operating and capital expenses by leveraging virtualization technology to design, deploy and manage network services. It was shown in that NFV effectively decouples the physical network equipment from the software that run on them. Other emerging enabling technologies that can solve the hard-to-reach problem include SDR, CR and RSMA and it has been shown that these can provide optimal low-cost connectivity solutions ,, and .

In order to extend coverage to hard-to-reach areas, looked into how new satellite-based services could be utilized to serve regions where conventional cellular coverage is absent. It discussed collaborative efforts by the Europe, Middle-East and African (EMEA) Satellite Operation Association (ESOA) to promote complementarity between terrestrial and non-terrestrial networks and encouraged the development of coverage-extending solutions. Also, analyzed and discussed the technologies suitable for extending coverage to rural, poor and isolated areas. It highlighted the impact of the choice of technology on the challenges of connectivity in hard-to-reach areas. It also proposed an architecture for future networks, based on the existing solution to eliminate the coverage gap.

For mobility management in IP-based networks, the Internet Engineering Task Force (IETF) has introduced a number of protocols, such as Mobile Internet Protocol version 6 (MIPv6) and Proxy Mobile Internet Protocol version 6 (PMIPv6). However, such protocols were not designed to deal with the high topology change rate as is common in LEO satellite constellations. A number of approaches have been proposed by to address this problem. Nevertheless, the concept of separating control plane and data plane of Software Defined Network (SDN) is a promising approach to efficiently manage the satellite network topology.

The integration of satellite and terrestrial networks brings unique security challenges due to the different system requirements and the incoherency of their security policies. Therefore, provides insights into some security concerns and provides some potential mitigation techniques. It was noted that the overall integrated system will have a higher degree of vulnerability and face higher security threats if technologies with weak security and loopholes are integrated together. In conclusion, l**ooking at the case studies reviewed in the literature, the related works did not take into cognizance the analyses of Nigerian scenario and this indeed is a research gap that this research work intends to fill.**

3. METHODOLOGY

The three standard orbits used by communication satellites are LEO (500 km–1,500 km), MEO (between LEO and GEO), and GEO (35,786 km). NigComSat-IR is in GEO and its orbital slot is at 42.5°E . This gives the satellite an average slant path of 42,164.21 km when viewed from its network control center in Abuja, Nigeria. This study adopts a Matlab-based modelling approach to uncover important parameters for integrating NigComSat-1R and other GEO satellites into a 6G network to bridge the digital divide in the country. The other GEO satellites used for the comparative analysis include EchoStar 16, located at 61.5°W, Astra, located at 5.2°E, and EchoStar 6, located at 72.7°W . Geospatial computations were conducted in Matlab to evaluate satellite visibility, and compute slant path distances. Link budgets were calculated and validated using ITU-R recommended models.

The study evaluated two primary use cases: **GEO-based low data rate for fixed Internet of Things (IoT) services** using Inmarsat NB-IoT standard **andLEO-based community WiFi backhaul** using OneWeb (Ku-band). The OneWeb use case assumes the utilization of flat panel antennas mounted on public or private infrastructure and interfaces with the ground network, with the aim of serving remote communities. The parameters include an orbital altitude of 1200 km, the service link utilizes 11.7GHz in the downlink and 14.5GHz for the uplink, and the satellites carry regenerative payloads . The study area was Nigeria's territorial space, with Uyo (Latitude: 5.024295°, Longitude: 7.872312°) used as the ground station reference point. Figure 3 shows the geometry of a GEO satellite.

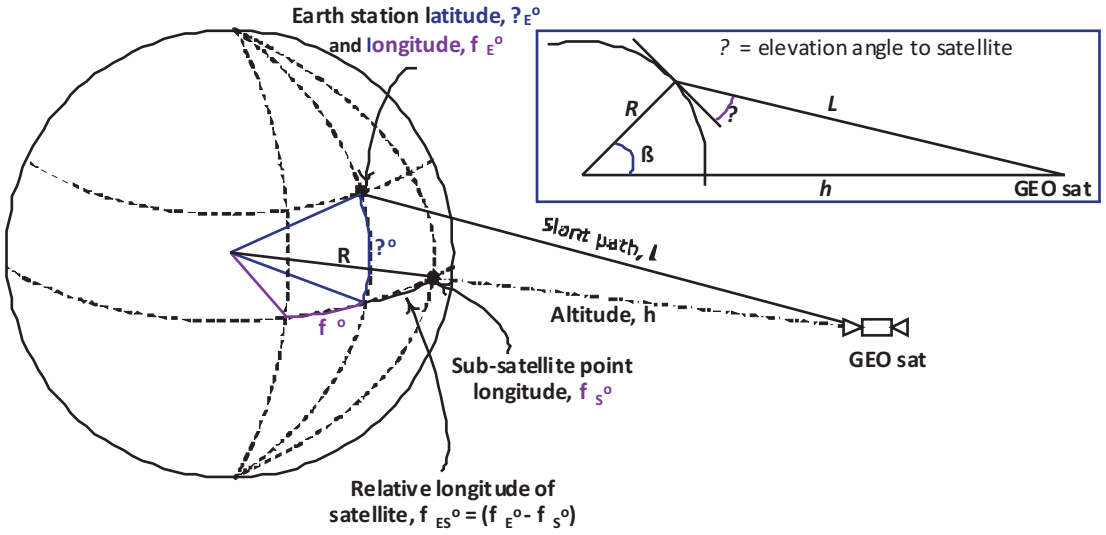


Figure 3: Geometry of a GEO satellite

As can be observed in Figure 3, the slant path, L, can be determined in terms of the sub-satellite point longitude,φ_S, the earth station longitude, φ_E, and earth station latitude, θ_E. The relative longitude of the satellite is therefore given as

$$\varphi_{ES} = \varphi_E - \varphi_S \tag{3.1}$$

Other important parameters are the satellite altitude, h, satellite elevation angle, ψ and the radius of the earth, R. The radius of the satellite is therefore given as

$$R_s = R + h \tag{3.2}$$

The satellite elevation angle is given as

$$\psi = \tan^{-1}\left[\frac{\cos\beta - \sigma}{\sin\beta}\right] \tag{3.3}$$

where $\beta = \cos^{-1}(\cos\theta_E \cos\theta_{ES})$, and $\sigma = \frac{R}{R+h}$ (3.4)

The slant path to the satellite is given as

$$L = 35{,}786\sqrt{1 + 0.4199(1 - \cos\beta)}\ \text{km} \qquad (3.5)$$

Having computed the slant path, it is necessary to determine the loss a signal suffers as it travels from the satellite to the hard to reach area. This is indicated as the free space loss and is given in decibels as

$$FSL = 20\,log\left(\frac{\lambda}{4\pi d}\right) \qquad (3.6)$$

As a function of frequency and slant path, the free space loss is given as

$$FSL = 32.45 + 20\log(f) + 20\log(L) \qquad (3.7)$$

In order for an earth station to be accessible to a satellite, the following condition must be satisfied.

$$\gamma \leq \cos^{-1}\left(\frac{R}{R_S}\right) \qquad (3.8)$$

In addition to free space loss, a propagating signal also suffers attenuation due to atmospheric effects, mainly due to hydrometeors such as rain, snow, and clouds. The long terms statistics of attenuation due to rain in the earth station-satellite slant path is dependent on the satellite elevation angle, latitude of the earth station, frequency of the propagating signal, height of the earth station above the mean sea level, h_s, and point rainfall rate where the earth station is located. The point rainfall rate, $R_{0.01}$ is computed for 0.01% of an average year [26]. For satelite elevation angels $\psi > 5^0$, the slant path range below the rain height is given by

$$L_s = \frac{h_R - h_S}{\sin\psi}\ \text{km} \qquad (3.9)$$

Where h_R is the effective rain height and is calculated based on the latitude of the earth station . For $\psi < 5°$, the following formula is used

$$L_s = \frac{2(h_R - h_S)}{(\sin^2\psi + 2(h_R - h_S)/R)^{1/2} + \sin\psi}\ \text{km} \qquad (3.10)$$

It is also important to consider the effects of shadowing, which is caused by large obstacles which absorb and attenuate the propagating signals. The effects of shadowing is described statistically, and it follows a log-normal distribution.

A link budget accounts for all the gains and losses from the transmitter output to the receiving antenna. For example, the NigComSat-1R C-band transponder has the following parameters: a bandwidth, of 36 MHz, an output power, $P_t = 20\,\text{W}$, a

satellite antenna gain, $G_t = 20$ dB, receiving earth station antenna gain, $G_r = 49.7$ dB, and a slant range as computed using (3.5) [23]. Therefore, according to [28], the power received by an earth station is given as

$$P_r = \frac{P_t G_t G_r}{(4\pi L/\lambda)^2} \quad (3.11)$$

The receiver noise power is given as

$$P = kT_s B_n \quad (3.12)$$

Where *k* is Boltzmann's constant, *Ts* is the system noise temperature, and B_n is the noise bandwidth. The carrier to noise ratio is therefore given as

$$\frac{C}{N} = \frac{P_r}{kT_s B_n} \quad (3.13)$$

The carrier to noise power ratio in the downlink is calculated based on the following equation

$$\frac{C}{N_{DL}} = EIRP_{SAT} + 20\,log\left(\frac{\lambda}{4\pi L}\right) + G_{RX} - 10\,log(kT_s B_n), \quad (3.14)$$

Where $EIRP_{SAT}$ is the effective isotropic radiated power of the satellite, G_{RX} is the gain of the antenna system in decibels, and other variables are as earlier defined in (3.12). Similarly, the uplink budget is calculated as follows.

$$\frac{C}{N_{UL}} = EIRP_{ES} + 20\,log\left(\frac{\lambda}{4\pi L}\right) + G_{RX} - 10\,log(kT_s B_n) \quad (3.15)$$

Satellite visibility parameters were derived using the standard geostationary-geometry model. The Earth's mean radius was taken as $R_e = 6371$ km and orbital altitude $h = 35786$ km, giving a satellite radius $R_s = R_e + h$. For a ground station at latitude θ_E, Longitude φ_E, and a satellite at sub satellite longitude φ_s, the elevation angle E is:

$$E = tan^{-1}\left[\frac{\cos\theta_E . \cos\varphi_s - R_e/R_s}{\sqrt{1 - [\cos\theta_E . \cos\varphi_s]^2}}\right] \quad (3.16)$$

The slant-range distance S and surface-coverage radius r_c were computed respectively as:

$$S = \sqrt{R_s^2 + R_g^2 - 2R_sR_g cos\psi} \quad (3.17)$$

$$\psi = cos^{-1}(cos\theta_E cos\ \varphi_s) \quad (3.18)$$

$$r_c = R_g cos^{-1}\left(\frac{R_g}{R_s}\cos E\right) \quad (3.19)$$

All trigonometric operations were performed in radians, and final outputs converted to degrees or kilometres as appropriate. The computations were verified in MATLAB (R2025b) and the elevation angles in Tables 3 and 4 agree, confirming geometric consistency between visibility and slant-path analyses. Figure 4 shows a screenshot of the MATLAB simulation environment.

The bandwidth for each link was chosen according to the standard channel spacing used in satellite IoT communication systems (200 kHz for downlink and 15 kHz for uplink, while spectral efficiency (η) was obtained from the selected modulation and coding scheme (QPSK with coding rates of 1/3 – 2/3, corresponding to 0.67–1.33 bps/Hz). These parameters were then applied in Equation (3.20), to compute the achievable data rates shown in Table 6.

$$R = \eta * B \quad (3.20)$$

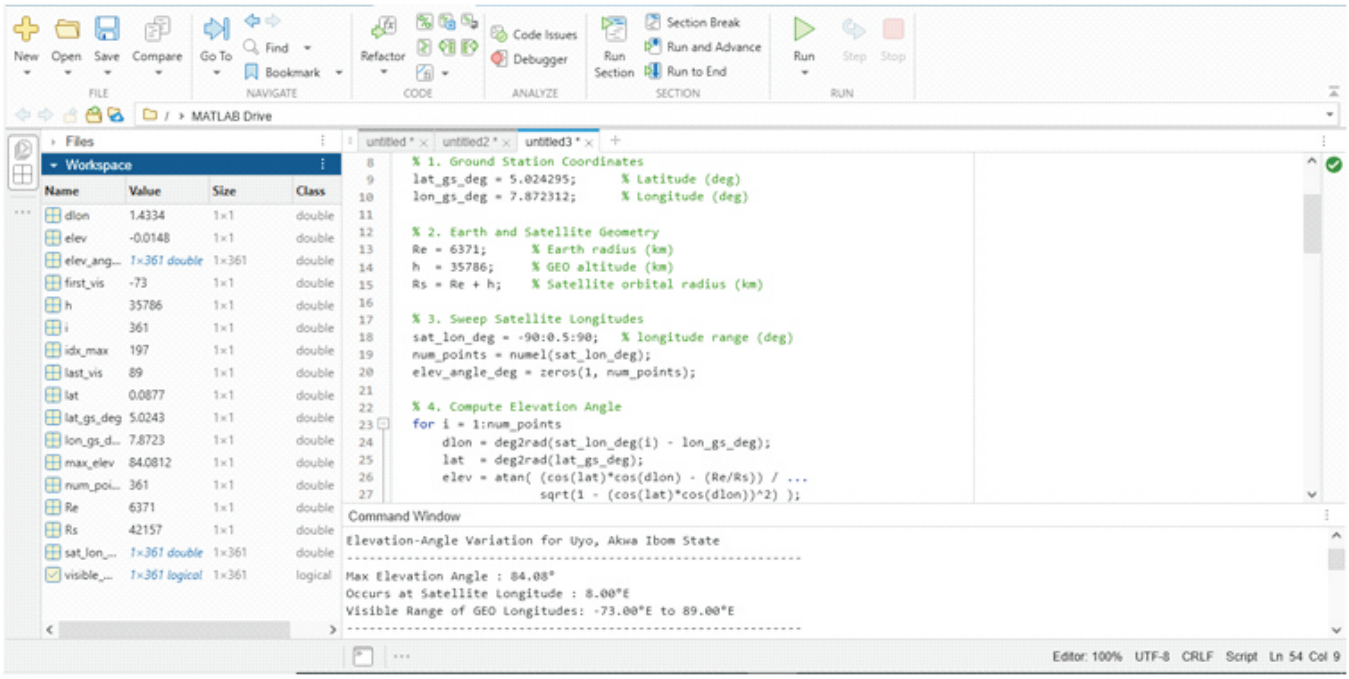


Figure 4: Matlab simulation screenshot

4. RESULTS AND DISCUSSION

The results are presented and discussed in terms of satellite visibility and slant path analysis, link budget analysis in C-band, and data use estimation for IoT use cases.

4.1. SATELLITE VISIBILITY AND SLANT PATH ANALYSIS

Satellite visibility was determined using the trigonometric models given in (3.1) to (3.5). As shown in Table 3, all four selected satellites had elevation angles greater than the 5° visibility threshold. The confirmation of line-of-sight connectivity implies that these satellites are viable candidates for delivering mobile communication services within the Nigerian territory.

Table 3: Satellite Visibility Results

S/N	Satellite Name	Satellite Longitude (°)	Earth Station Lat (°)	Earth Station Long (°)	Elevation Angle (Computed)	Visibility Status
1	NigComSat-1R	42.4	5.024295	7.872312	38.2	Visible
2	EchoStar 16	-61.5	5.024295	7.872312	41.7	Visible
3	Astra	5.2	5.024295	7.872312	46.5	Visible
4	EchoStar 6	-72.7	5.024295	7.872312	40.2	Visible

Table 4: Slant Path Distance Results

Satellite Name	Slant Range (km)	Coverage Radius (km)	Elevation Angle (°)
NigComSat-1R	38 404.7	8 170.5	38.21
EchoStar 16	37 956.8	8 546.1	41.73
Astra 1A	37 221.3	9 119.4	46.51
EchoStar 6	38 709.9	7 968.7	40.28

The slant path distances are shown in Table 4 and the values represent the geometric line-of-sight distances between satellite and ground station. The distances serve as critical input in link budget calculations, directly influencing free space path loss and, by extension, signal quality. The relationship between satellite elevation angle and slant path distance is shown in Figure 5.

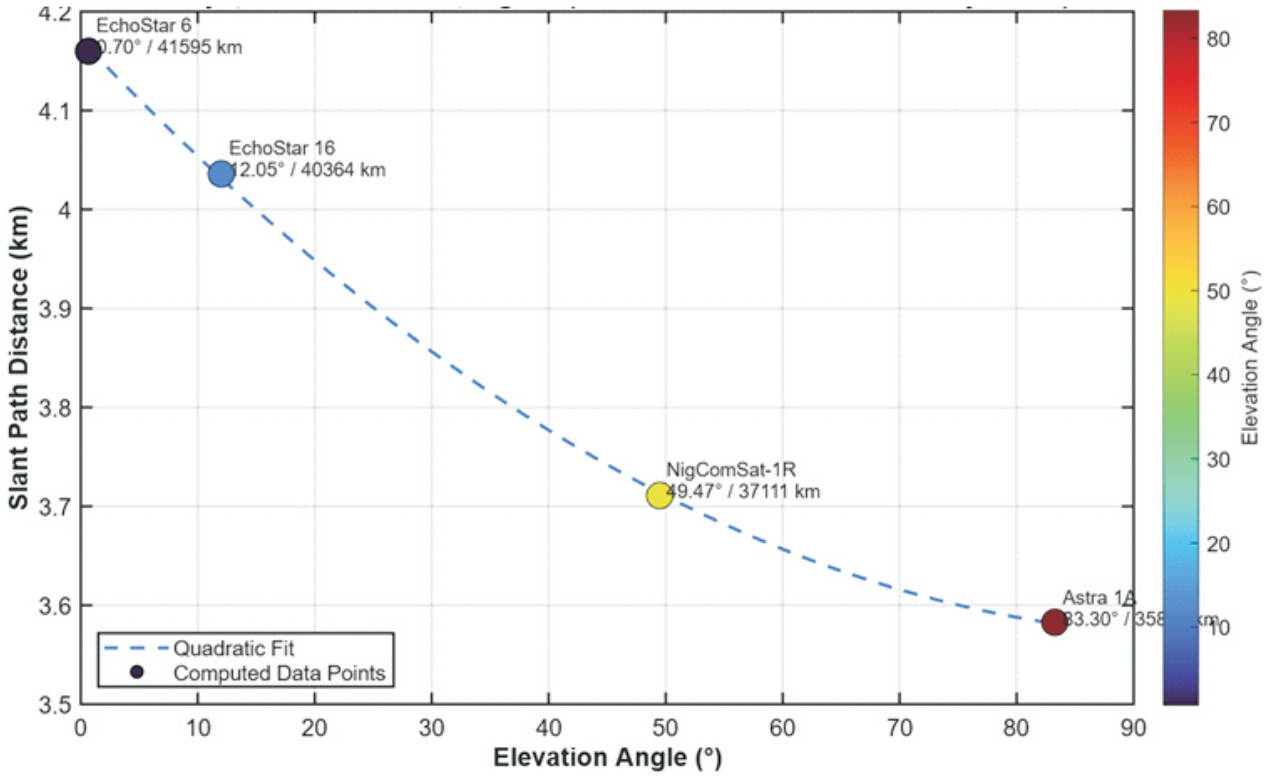


Figure 5:*Relationship between elevation angle and slant path*

4.2. LINK BUDGET ANALYSIS FOR NIGCOMSAT-1R (C-BAND)

As shown in Table 5, the downlink budget from NigComSat-1R to a typical Class 3 IoT device showed a C/N ratio of 20.36 dB, which is adequate for narrowband data transmission. The low receiver G/T of -3 dB/K reflects the limitations of IoT-class devices, but with sufficient EIRP and minimal atmospheric loss, the signal remains viable. The uplink results demonstrates a superior C/N ratio of 24.61 dB. This value is made possible by the satellite's high G/T (11 dB/K) and the manageable free space and atmospheric losses. The uplink performance exceeds downlink, highlighting the satellite's enhanced reception capabilities. Figure 6 shows the signal level diagram for the uplink and downlink paths.

Table 5: Link Budget

Parameter	Downlink	Uplink
EIRP (dBW)	44	23.0
Free Space Loss (dB)	195.73	195.73
Atmospheric Loss (dB)	0.5	0.5
Receiver G/T (dB/K)	-3.0	11.0
C-band Bandwidth (GHz)	200	36
C/N Ratio (dB)	20.36	24.61

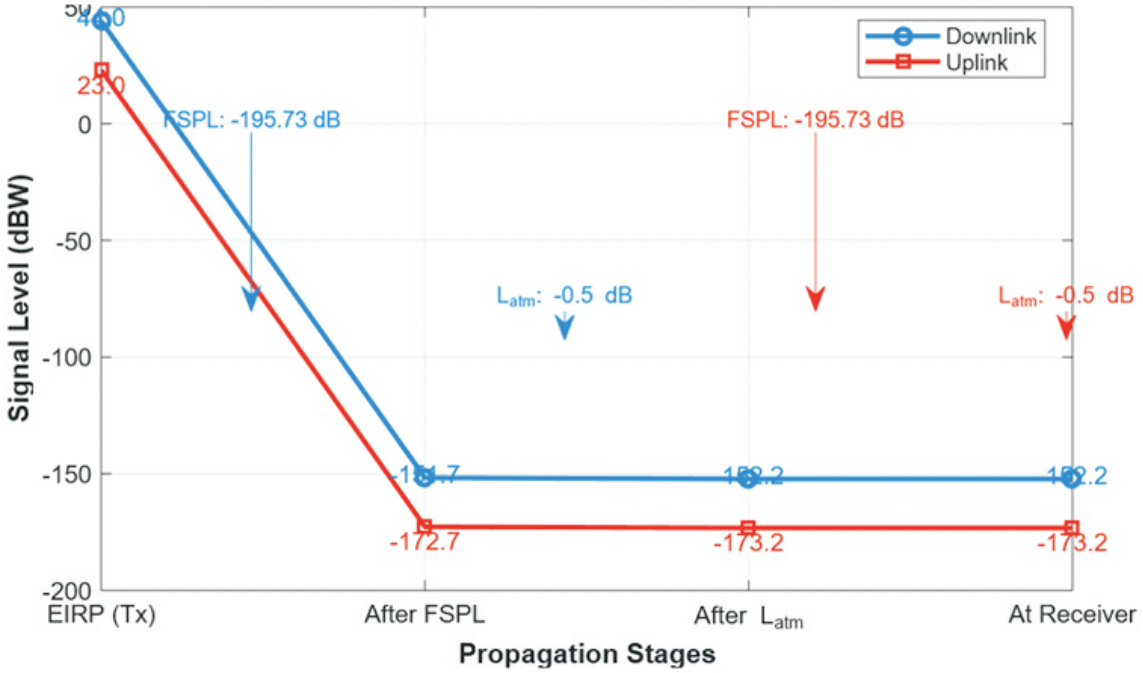


Figure 6: *Signal level diagram for uplink and downlink paths*

4.3. DATA RATE ESTIMATION FOR IOT USE CASE

Table 6 shows the data rates that NigComSat-1R can support for basic IoT services like smart metering and environmental monitoring. The configuration adopted nearly doubles the downlink data rate and delivers acceptable uplink rates for intermittent sensor data transmissions.

Table 6: Data Rates for IoT Devices

Link Direction	Bandwidth (kHz)	Spectral Efficiency (bps/Hz)	Data Rate (kbps)
Downlink	200	0.67	134.0
Uplink	15	0.67	10.05
Downlink (Opt.)	200	1.33	266.0
Uplink (Opt.)	3.75	1.60	6.0

4.4. THROUGHPUT ANALYSIS FOR ONEWEB (LEO, KU-BAND)

As shown in Table 7, the OneWeb LEO scenario achieved very high throughput, supporting real-time and bandwidth-intensive applications like community WiFi or cellular backhaul. The range between best and worst-case scenarios highlights the dependency on terminal G/T and satellite elevation but confirms the system's flexibility in serving variable terrains and user densities.

Table 7: Data Rates for Community WiFi via LEO Constellations

Link Direction	Scenario Type	G/T (dB/K)	Data Rate (Mbps)
Downlink	Best Case	9	830
Downlink	Worst Case	7	140
Uplink	Best Case	9	880
Uplink	Worst Case	7	140

5. ENGINEERING IMPLICATIONS

From an engineering standpoint, the findings validate the potential of satellite systems to extend mobile communication to hard-to-reach areas. The slant path and elevation angle data confirm that GEO satellites can provide consistent coverage. The link budget calculations affirm the feasibility of stable communication links for narrowband applications. Furthermore, the high data rates demonstrated in the OneWeb case, makes LEO satellites suitable for backhaul and fixed wireless access in underserved regions. These insights are critical for engineers designing future 6G NTN-terrestrial hybrid systems.

6. CONCLUSION AND RECOMMENDATIONS FOR FURTHER WORK

This paper has presented a detailed engineering analysis of satellite-enabled 6G network communication systems as a means of addressing Nigeria's mobile coverage gaps in hard-to-reach areas. The work confirmed satellite visibility, computed feasible slant paths, developed and analyzed link budgets, and modeled real throughput for IoT and broadband use cases. Both GEO and LEO satellites proved capable of supporting varying classes of services, thus offering scalable solutions to connectivity challenges. Future work may integrate NS-3 dynamic simulations to explore delay, jitter, and multi-hop satellite-terrestrial routing performance under real-world conditions.

REFERENCES

[1] R. Adeleke, "Digital divide in Nigeria: The role of regional differentials," *African Journal of Science, Technology, Innovation and Development,* 2020.

[2] U. M. Ekpe, V. B. Umoh and N. S. Agbeb, "Eliminating the Digital Divide in Nigeria: Policy Direction and 5G Deployment Methodology," in *2021 1st International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS)*, Abuja, Nigeria, 2021.

[3] X. Zhu and C. Jiang, " Integrated Satellite-Terrestrial Networks Toward 6G: Architectures, Application," *IEEE Internet of Things Journal,* vol. 9, no. 1, pp. 437-461, 2022.

[4] W. Abderrahim, O. Amin, M.-S. Alouini and B. Shihada, "Latency-Aware Offloading in Integrated Satellite Terrestrial Networks," *IEEE Open Journal of the Communications Society,* vol. 1, pp. 490-500, 2020.

[5] F. Ulebor, "Nigeria has achieved 79.65% teledensity, 48.81% broadband penetration – NCC," 17 July 2025. [Online]. Available: https://www.vanguardngr.com/2025/07/nigeria-has-achieved-79-65-teledensity-48-81-broadband-penetration-ncc/. [Accessed 1 November 2025].

[6] S. Yrjola, M. Matinmikko-Blue and P. Ahokangas, "The Evolution of Mobile Communications," in *The Changing World of Mobile Communications*, Palgrave Macmillan, 2023, pp. 13-43.

[7] Satnews, "Eutelsat and NIGCOMSAT's multi-year, multi-million-dollar partnership," 23 January 2025. [Online]. Available: https://news.satnews.com/2025/01/23/eutelsat-partners-with-nigcomsat/. [Accessed 1 November 2025].

[8] U. M. Ekpe, A. L. Imoize and W. Montlouis, "Massive MIMO for Non-terrestrial Wireless Communication Systems," in *Massive MIMO for Future Wireless Communication Systems: Technology and Applications*, John Wiley & Sons, 2024, pp. 371-402.

[9] U. M. Ekpe, "RIS-Empowered Terrestrial and Non-terrestrial Wireless Communication Systems," in *Reconfigurable Intelligent Surfaces for 6G and BEyond Wireless Networks*, Wiley, 2025, pp. 525-550.

[10] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar and J. F. M. Montoya, "Satellite Communications in the New Space Era: A Survey and Future Challenges," *IEEE Communications Surveys & Tutorials,* vol. 23, no. 1, pp. 70-109, 2021.

[11] World Bank, "World Bank Open Data - Rural Population," 2023. [Online]. Available: https://data.worldbank.org/indicator/SP.RUR.TOTL.ZS. [Accessed 1 November 2025].

[12] U. M. Etian and P. T. Bemgba, "Banditry in Northern Nigeria: Terrorists or Marauders?," *International Journal of Social Science, Management, Peace and Conflict Research,* vol. 9, no. 1, pp. 96-109, 2025.

[13] A. A. Felix, F. M. Dahunsi and S. O. Oluwatoki, "A Systematic Review on the Quality of Service of the Nigerian Communication System," *International Journal of Advances in Engineering and Management,* vol. 6, no. 8, pp. 199-212, 2024.

[14] I. S. Tognise, J. Degila and A. D. Kora, "Connecting Rural Areas: A Solution Approach to Bridging the Coverage Gap," in *IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference*, New York, 2021.

[15] T. F. Collins, R. Getz, D. Pu and A. M. Wyglinski, Software-Defined Radio for Engineers, Artech House Publishers, 2018.

[16] B. Clerckx, Y. Mao, E. A. Jorswieck, J. Yuan, D. J. Love and E. Erkip, "A Primer on Rate-Splitting Multiple Access: Tutorial, Myths, and Frequently Asked Questions," *IEEE Journal on Selected Areas in Communications,* vol. 41, no. 5, pp. 1265-1308, 2023.

[17] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. D. Turck and R. Boutaba, "Network Function Virtualization: State-of-the-Art and Research Challenges," *IEEE Communications Surveys & Tutorials,* vol. 18, no. 1, pp. 236-262, 2016.

[18] T. Darwish and G. K. Kurt, "LEO Satellites in 5G and Beyond Networks: A Review from a Standardization Perspective," *IEEE Access,* vol. 10, pp. 35040 - 35060, March 2022.

[19] M. A. Abir, M. Z. Chowdhury and Y. M. Jang, "Software-Defined UAV Networks for 6G Systems: Requirements, Opportunities, Emerging Techniques, Challenges, and Research Directions," *IEEE Open Journal of the Communications Society,* vol. 4, pp. 2487-2547, 2023.

[20] A. Khan, M. Jaffar, S. Yaghmour, S. Watts, N. Chuberre and S. Mugnaini, "Extreme Long-Range Communications for Deep Rural Coverage: Non-Terrestrial Networks Position Paper," NGMN Alliance, 2019.

[21] F. T. Zohra, S. Azam and M. M. Rahman, "Overview of IPv6 Mobility Management Protocols and their handover," *International Journal of Computer Sciences and Engineering,* vol. 2, no. 3, pp. 121-129, 2014.

[22] I. Ahmad, J. Suomalainen and P. Porambage, "Security of Satellite-Terrestrial Communications: Challenges and Potential Solutions," *IEEE Access,* vol. 10, pp. 96038-96052, September 2022.

[23] NigComSat, "NigComSat-IR Technical Manual 2012 Edition," NIGCOMSAT, Abuja, 2012.

[24] E. Johnston, "List of Satellites in Geostationary Orbit," 24 October 2025. [Online]. Available: https://www.satsig.net/sslist.htm.

[25] U. M. Ekpe, *Satellite Communication Fundamentals Lecture Notes,* Ikot Akpaden: Department of Electrical and Electronic Engineering, Akwa Ibom State University, 2023.

[26] ITU-R, "Propagation Data and Prediction Methods Required for the Design of Earth-Space Telecommunications Systems," International Telecommunications Union, Geneva, July 2010.

[27] P. T. Thompson, "Propagation considerations relating to satellite communication systems," in *Satellite Communication Systems, 3rd Ed*, London, The Institution of Electrical Engineers, 2000, pp. 99-115.

[28] B. G. Evans, "Satellite systems planning," in *Satellite Communication Systems, 3rd Ed*, London, The Institution of Electrical Engineers, 2000, pp. 199-222.

A comprehensive analysis of multi-stage cyber-attack detection and prevention Using Machine Learning approaches: A review.

Authors: Akpasam J. Ekanem, Department of Electrical and Electronic Engineering, Akwa-Ibom State University, Ikot-Akpaden. akpasamekanem@aksu.edu.ng

AND

Okon Nsaufot, Department of Computer Engineering Technology, Akwalbom State Polytechnic, Ikotosurua, ufot.okon@akwaibompoly. edu.ng.

Abstract

The increasing sophistication of cyber threats has made multi-stage cyber-attacks a critical concern for modern networked systems. Unlike single-stage attacks, multi-stage attacks involve a sequence of coordinated actions that evolve over time, making their detection significantly more complex. This paper presents a comprehensive review of Machine Learning-based approaches for the detection and prevention of multi-stage cyber-attacks. The study categorizes existing techniques into supervised, unsupervised, semi-supervised, and reinforcement learning paradigms, and evaluates their effectiveness based on detection accuracy, adaptability, and computational efficiency. Furthermore, this review highlights key challenges such as data imbalance, lack of standardized datasets, high false positive rates, and limitations in detecting zero-day attacks. A comparative analysis of existing models is presented to identify research gaps and performance trade-offs. Finally, future research directions are proposed, emphasizing the need for hybrid intelligent systems, improved datasets, and advanced learning frameworks capable of handling evolving attack patterns. This study provides valuable insights for researchers and practitioners aiming to design robust and scalable cyber defense systems.

1. **Introduction:** Cybersecurity is a major concern for countless organizations, institutions, corporations, and individuals globally. Buczak and Guven [1] provide a precise definition of cyber security as the encompassing array of technologies and methodologies employed to oversee and thwart unauthorized entry, modification, abuse, and disruption of computer networks and resources. This also includes the ability to control and authorize access to sensitive information and critical infrastructure that can be reached through a network.

Most networks are highly interconnected through the Internet, enabling the interchange of data, information, knowledge, software, and hardware. The computer networking paradigm has enabled the exchange of crucial resources to enhance operational efficiency. However, it has also facilitated the widespread dissemination of malware, resulting in an increase in cyber-attacks in the digital domain.

The expansion of potential risks is a consequence of the growing impact of cyber capabilities, which are gradually penetrating and impacting all parts of home, commercial, and industrial processes. Akyazi in[2] asserts that cyber-attacks pose a risk by virtue of their ability to modify system or database parameters, which can have a kinetic effect that may increase the attacks and perhaps result in the destruction of confidential information.

To protect against cyber-attacks, it is essential to employ both proactive and reactive tactics. The strategies, known as active and passive, are relevant in the given context of application. They encompass proactive defense measures or mitigation strategies against cyber threats. Denning [3] argues that the significance of cyber defense measures resides in their capacity to efficiently counteract both active and passive threats, which have become widespread in the cyber realm.

The differentiation between single-stage attack detection and multi-stage attack detection is substantial. A single-stage attack effectively exploits the target system by carrying out simple and indiscriminate attack attempts within a short period of time. Nevertheless, because of the repetitive and indiscriminate manner in which the single-stage assault is carried out, it rapidly creates a track of evidence that can be readily identified by most inline or endpoint protection systems. Multi-stage attacks differ from single-stage assault detection in that they employ a more intricate and protracted offensive approach when compared to single-stage attacks. For example, to circumvent the standard security measures, the time intervals of the multi-stage attack range from a few minutes to several months. To properly detect and respond to the multi-stage attack, the administrator must carefully monitor and correlate individual attack alerts from different machines and attack scenarios, even if identifying each specific attack stages are not very difficult. Hence, identifying multi-stage attacks is extremely difficult without previous awareness of such instances.

Understanding the research gaps in current cyber security approaches is crucial. The present study will analyse theMachine Learningapproaches available in the public realm, clarifying the advantages and disadvantages of each approach. The next parts will analyse cyber security strategies in terms of detection, prevention, and challenges.

Attack detection and prevention can be achieved using a range of methods, such as Machine Learning and evolutionary algorithms, statistical techniques, association rules, similarity-based approaches, causal correlation, structural-based approaches, case-based approaches and mixed approaches [4,5,6]. Similarly, most strategies employed to thwart attacks include scrutinizing network data to identify and eradicate (or limit) malicious activities. This study will focus on Machine Learning approaches of multi-stage.Figure 1 depicts this.

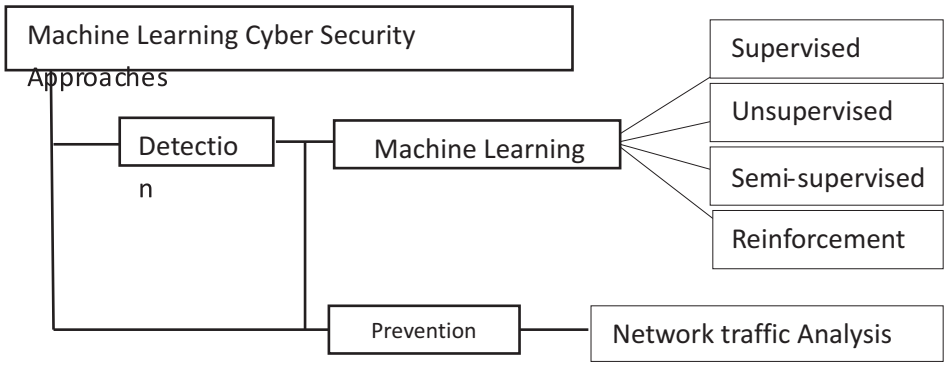


Figure 1: Machine Learningcyber security approaches

2. Methodology of Review

This study adopts a structured literature review methodology to analyze existing research on multi-stage cyber-attack detection and prevention using machine learning approaches.

2.1 Data Sources and Search Strategy

Relevant literature was collected from reputable academic databases, including IEEE Xplore, SpringerLink, ScienceDirect, and Google Scholar. These sources were selected due to their extensive coverage of peer-reviewed journals and conference proceedings in cybersecurity and machine learning.

A combination of keywords was used to retrieve relevant studies, including *"multi-stage cyber-attacks," "intrusion detection systems," "machine learning in cybersecurity," "anomaly detection,"* and *"zero-day attacks."* Boolean operators such as AND and OR were used to refine the search results.

2.2 Inclusion and Exclusion Criteria

To ensure the quality and relevance of the reviewed studies, the following inclusion criteria were applied:

- Studies focusing on multi-stage cyber-attack detection
- Research involving machine learning or hybrid approaches
- Peer-reviewed journal articles and conference papers
- Studies with clearly defined methodologies and evaluation metrics

The exclusion criteria included:

- Non-peer-reviewed articles and grey literature
- Studies not related to cybersecurity or intrusion detection
- Papers lacking sufficient experimental or methodological detail

2.3 Time Range of Selected Studies

The review focuses on studies published between **2013 and 2023**, capturing recent advancements in machine learning and cybersecurity. Earlier foundational works were also included where necessary to provide theoretical background.

2.4 Data Extraction and Analysis
Selected studies were carefully analyzed based on key parameters, including methodology, dataset used, performance metrics, contributions, and identified limitations. The extracted information was synthesized to compare different approaches and highlight research gaps in multi-stage attack detection.

2.5 Limitations of the Review Methodology
Although this review follows a structured approach, it is limited by the availability of publicly accessible datasets and published studies. Additionally, variations in evaluation metrics and experimental setups across different studies make direct comparison challenging.

3. Review of Existing Literature
This section focuses on the detection of multi-stage cyber-attacks using machine learning approaches.

3.0.1 Detection using Machine LearningAlgorithms
Machine Learning methods have been increasingly popular in recent years for identifying cyber-attacks. Machine Learning is a powerful tool for examining data and making accurate predictions about the outcomes of certain events. It accomplishes this by employing sample inputs to create an appropriate model that enables precise decision-making [1]. Classifying and predicting the presence or absence of a given sample using training data is the primary objective of Machine Learning algorithms. The application of Machine Learning in the current context of cyber-attack detection has greatly improved the precision of the detection process.

This study examines four discrete Machine Learning methodologies. The strategies include supervised, unsupervised, semi-supervised, and reinforcement learning.

I. Supervised Learning
Supervised or supervised learning refers to an educational method that incorporates the active involvement and guidance of a supervisor or mentor. Supervised learning is a subset of pattern recognition that employs a set of labelled instances, referred to as training data, together with their corresponding desired output. During the training phase, a predictive model is constructed using the labelled cases to classify fresh datasets. This is achieved by feeding the cases that have been assigned labels into a designated Machine Learning algorithm. Machine Learning techniques covered in [1] include Artificial Neural Networks, K-Nearest Neighbour (KNN), Support Vector Machines (SVM), Hidden Markov Models (HMM), Decision Trees and Naïve Bayes.

a. Decision Trees: This algorithm categorizes events based on the values of the feature. Each feature in a classified event is represented by a node in the model, and the branches represent the possible values for that feature. Events are categorized beginning from the top node and organized according to their feature values. At each level of the decision tree, the algorithm selects the feature that best separates the

events into subclasses, using methods such as entropy or information gain[7].

b. Support Vector Machine (SVM): The SVM translates data into higher dimensions and identifies the optimum hyper-plane for data separation. It employs the concept of margin to determine the maximum margin of the dataset. The concept of a “margin” with carrying sides that separate two data classes is central to SVMs. It increases the margin, resulting in the greatest feasible separation between separating hyper-planes [7].

c. Artificial Neural Network: Neurons are used to create a neural network that represents the human brain [8]. It has hidden layers that can perform data processing and transfer the output to the output layer. A pattern that is generated from the data is used to train the Neural Network (NN). The output is retrieved and checked, and if it is correct, the next pattern is supplied as input. When a mistake occurs, there is an error that is propagated backward to the input layer (backpropagation algorithm), the weights are adjusted to obtain the output for all training patterns [9]. Some of the **advantages of Artificial Neural Network includes the** ability to comprehend and stimulate complex and nonlinear relationships, ability to generalize the model and anticipate data that have yet to be observed and partially resistant to harm [10]. However, due to the enormous number of parameters to be set, optimizing the network might be difficult and large neural networks require significant computational time.

d. Random Forest: this classifier is an ensemble classifier that makes predictions. A portion of the training dataset is selected via substitution to construct trees (a bagging approach). In other words, certain samples can be used several times, and others may never be chosen. The model requires defining the number of decision trees (Ntrees) and the number of variables to be selected for optimal splitting.[11, 12]. The strength of Random Forest is that it enhances the accuracy of classification and works well with datasets with a large number of input variables[10]. However, Random Forest is quick to train, but it is slow to make predictions once it has been trained. Also, the evaluation is time consuming and interpretation is difficult [10].

e. XGBoost: this model is based on decision tree ensemble Machine Learning system that applies a gradient boost algorithm to a known dataset before classifying it. The execution speed and the model's execution are two of the benefits of using XGBoost. When XGBoost is compared with other gradient-boosting implementations, the results reveal that it is quicker [13].

f. K-Nearest Neighbour: The K-nearest neighbor technique works by classifying a fresh data utilizing previously classified data. A test sample refers to data that is uncertain about its class, while a training sample is data that has already been categorized. KNN approach determines the test sample and the training sample distances and it selects the k-nearest training samples that are most similar to the

test sample. If the majority of these selected k samples belong to a specific class, the test sample is assigned the same. [14]. **k-Nearest-Neighbour is easier to deploy and** training is completed more quickly. The **disadvantages of the K-Nearest Neighbour is that** finding the nearest neighbor in the training data, which is massive, takes a long time, making it slow. Additionally, it requires large storage and lacks transparency in the representation of knowledge.

g. Naïve Bayes: The model is a statistically driven classification system that assigns labels to data. It is commonly employed in categorization tasks due to its simplicity. In Bayesian classification, the goal is to compute the conditional probability of the class to which the data belongs, aiming to estimate the probability of a class based on the provided data [15]. **Naïve Bayes is simple to use** and the outcome is more accurate due to the higher probability value but there is a strong assumption about the form of data distribution and there is a loss of precision.

h. Logistic Regression (LR): This model determines the link between a large number of variables that are either independent or dependent. LR is now used in social science because of the inefficiency of the least squats method (LSM) in a multivariate model with discrimination of dependent and independent variables. In LR, prediction is based on the chances associated with the two possible values of the dependent variable [15]. In **logistic Regression, continuous outco**mes are impossible to forecast, it is susceptible to overconfidence (i.e., the models may appear to have greater predictive potential than they do, leading to overfitting). Also, to get consistent findings, a large sample size is needed [10].

Osarumwense et al. [16] developed a probabilistic inference method to anticipate multi-stage attacks originating from malicious IP addresses, employing a supervised Machine Learning technique referred to as a Causal Network, or Bayesian Belief Network (BBN) Model. This approach forecasts multi-stage attacks by utilizing a joint probability density function, facilitating the creation of a Bayesian attack graph. This graph assigns particular probabilistic values to each attack and attack mode, enabling the calculation of the likelihood of both single-stage and multi-stage attacks. This model is engineered for use in computer network infrastructures, offering essential information to bolster network protection through enhanced prediction and detection of multi-stage attacks, blacklisting of malicious IPs, and overall improvement of network security.

The system exhibits considerable progress in multi-stage attacks prediction and detection; yet, it possesses limitations. It fails to anticipate multi-stage attacks employing MAC addresses or devices utilizing VPNs, it cannot also detect zero-day attacks, potentially creating deficiencies in its comprehensive security coverage. The Bayesian Belief Network model serves as an essential instrument in enhancing defenses against progressively intricate cyber dangers.

Verkerken et al. [17] have suggested a revolutionary multi-stage approach for hierarchical intrusion detection, demonstrating substantial progress in network security. The experimental solution was meticulously assessed utilizing the CIC-IDS-2017 and CSE-CIC-IDS-2018 network intrusion datasets, exhibiting significant adaptability without necessitating the retention of any classifiers. This adaptability facilitates an n-tier deployment approach, significantly minimizing bandwidth and processing demands while preserving the ability to identify zero-day assaults. This method prioritizes the preservation of privacy throughout the training and operational phases of hierarchical deployment. This aspect is becoming increasingly essential as firms strive to protect sensitive data while executing effective security protocols. The results demonstrate that several current models trained on network Intrusion Detection System (IDS) datasets frequently exhibit insufficient generalization capabilities, hence constraining their efficacy in practical applications.

The experimental execution of this multi-stage methodology depends exclusively on network characteristics as input; nevertheless, the proposed architecture is sufficiently adaptable for application in host-based and hybrid IDS environments. This versatility is essential for firms pursuing complete security solutions in diverse situations.

The innovative multi-stage method adeptly reconciles the trade-off between attaining high recall for zero-day attacks, minimizing bandwidth consumption, and preserving classification efficacy. The outcomes are remarkable, attaining a weighted F1 score of 0.9875 and a balanced accuracy score of 0.9342. The advanced method for multi-stage intrusion detection achieved scores of 0.9383 and 0.8550, underscoring the superior efficacy of Verkerken et al.'s approach. This research highlights the potential for novel strategies to improve intrusion detection systems, especially in responding to rising threats while maximizing resource efficiency.

The study by [18] presents a new Kill Chain State Machine (KCSM) aimed at improving the detection of multi-stage attacks through the analysis of clustered alert data. This novel approach markedly decreases the quantity of warnings by means of efficient alert correlation and attack contextualization, which is essential in cybersecurity, where analysts frequently encounter an excessive volume of notifications.

Analysts may now triage events using multi-stage scenario graphs produced by the KCSM algorithm, rather than sifting through hundreds of thousands of singleton alerts. The system uses correlated meta-alerts in conjunction with unclustered single alerts to create Advanced Persistent Threat (APT) scenario graphs, offering critical context about potential multi-stage attacks within the network. This contextualization allows analysts to concentrate on the most pertinent warnings, optimizing the incident response process.

The algorithm exhibits significant efficacy in reducing alerts. In a simple configuration, it produced 642 alerts from an original total of 12,735 alerts over a ten-day span, resulting in a drop of merely 5.04%. In a high-security configuration, the system diminished an astounding 446,458 alerts to merely 700, achieving a notable reduction of 0.16%. A reduction of two to three orders of magnitude in alerts results in a manageable amount for human analysts, hence enhancing the efficiency of threat identification and response.

The KCSM approach's primary feature is its flexibility, as it formulates stage deductions based on network direction without necessitating specific information inherent in the underlying alerts. This functionality enables the algorithm to be widely adaptable across diverse network-based alert systems. Furthermore, it is capable of processing additional stage-specific warnings, integrating the results into the APT scenario graphs to enhance the contextual comprehension of the attack. Nonetheless, it is important to highlight that the algorithm presently handles just network-level information and is incapable of including host-level and user identity circumstances. This constraint may limit the complexity of the created scenario graphs. The viability and efficacy of the KCSM technique were assessed through various experiments utilizing the CSE-CIC-IDS2018 dataset, validating its promise as a robust instrument for enhancing the analysis and response to multi-stage cyber threats.

ii. Methodology for Unsupervised Learning

Unsupervised learning entails the detection of patterns in a dataset that lacks any form of labelling. Subsequently, these patterns are employed to provide precise classification determinations for novel occurrences. Usually, this technique involves using clusters to identify the categories that examples belong to. In their study, Song et al. [19] investigated an anomaly detection system that employed unsupervised learning to automatically adapt and optimize parameter values. This was done to enhance the system's ability to classify events as either attack strings or normal connections.

The suggested methodology conducts instance classification following the training phase, which encompasses activities such as filtering, clustering, and modelling. Filtering is employed to extract the essential subset of regular data, which is then partitioned into k clusters. The k clusters represent common patterns observed in the traffic data, including HTTP, SMTP, and FTP. Every typical cluster undergoes the one-class SVM algorithm, which produces k-SVM models, also referred to as k-hyper-spheres, for classification. Subsequently, each k-model is compared with new instances to assess whether the instance lies within the predetermined hyper-sphere. If a connection satisfies certain criteria, it is categorized as a normal connection; otherwise, it is designated as an attack state.

By employing unsupervised learning in this approach, a highly effective technique is used to classify new instances by setting a threshold to distinguish between malicious and regular data throughout the model's development. Presently, a significant drawback of the approach is readily apparent since typical connections vary across different networks, which can considerably hinder the creation of precise profiles of normal behaviour. The significant variation in the behavioural patterns and characteristics of one network compared to other networks might lead to an inefficient model, requiring extensive parameter adjustment and optimization to match the specific network environment.

According to Abduvaliyev et al., [20] and Butun et al., [21] investigated several forms of attacks on wireless sensor networks (WSNs) and the potential influence of attack detection systems on the expanding threat landscape. However, as stated by [20], additional actions must be taken to protect a Wireless Sensor Network (WSN) from various forms of attacks, including denial of service (DoS), sinkhole/blackhole attacks, selective forwarding, node replication attacks, and wormhole attacks, in order to minimize their harmful consequences. The second line of defense utilizes a clustering technique that is categorized as an unsupervised learning process. This technique facilitates the detection of anomalous traffic in Wireless Sensor Networks (WSNs). The model is constructed by utilizing twelve network traffic patterns, which are subsequently employed throughout both the training and testing phases.

During the phase of training the model, a technique called fixed-width clustering is used to create clusters in the feature space. Anomalies in clusters are identified when the samples being analysed have a reduced training size.

Anomalies are detected when traffic samples exceed a specific threshold. In addition, the testing phase involves the identification of anomalous patterns by linking certain traffic samples with a cluster set. A significant drawback of this technology is the considerable processing requirements imposed on sensor nodes, resulting in major additional expenses for the main network.

Aparicio-Navarro et al. [22] presented a novel intrusion detection system (IDS) that utilizes contextual information, particularly pattern-of-life (PoL) data, to evaluate anticipated network behavior. This IDS is engineered to identify multi-stage assaults (MSA) in real time without any prior training. Findings demonstrate that the integration of contextual information markedly enhances the system's efficiency, increasing the detection rate of MSAs by 58% in real-time situations. This detection method enhances an unsupervised, anomaly-based IDS framework by adding a fuzzy cognitive map (FCM) to incorporate contextual information into the detection process.

The results demonstrate that the use of contextual data significantly improves the efficacy of Intrusion Detection Systems in detecting Malicious Software Activities.

The incorporation of an FCM resulted in a 58% enhancement in the detection rate (DR) relative to the IDS lacking this component. Nonetheless, the architecture of the FCM is exceedingly context-dependent, constraining its generalizability. A 5-step MSA was specifically designed for testing in the study, with the FCM adapted for this context. Thus, implementing the model in different situations or MSAs necessitates the creation of a new FCM tailored to those circumstances. The mechanism employed to capture temporal correlations among MSA phases is not transferable to other forms of multi-stage assaults. The temporal parameters established in this work are context-specific, and the IDS is incapable of autonomously adapting its detection methodology for differing MSAs.

Shin et al. [23] introduced an innovative architecture for multi-stage attack detection, comprising two primary phases: the creation of detection rules and the actual detection phase. This framework functions by creating precise detection rules designed for multi-stage assaults and subsequently evaluating incoming network data against these criteria. In contrast to conventional approaches, it does not necessitate prior knowledge of single-stage assault behaviors, rendering it adaptive and proficient in detecting diverse multi-stage attack patterns even in the absence of specific information on individual attack stages.

The framework exhibited robust performance when assessed, even when processing substantial amounts of intricate multi-stage attack data. It precisely recognized all multi-stage attack patterns inside the DARPA LLS DDoS dataset, validating its efficacy in a demanding test environment. Furthermore, in evaluations utilizing the CTU-13 datasets, characterized by extensive sophisticated assault patterns, the framework attained a peak F1 score of 0.9380, signifying a substantial degree of precision and recall.

The study emphasizes that a considerable percentage of network attacks are multi-stage, characterized by coordinated and intricate series of operations that occur over prolonged durations. These attacks are methodically divided into several discrete single-stage operations, rendering them challenging to identify when examined independently. Thus, recognizing these multi-stage patterns is crucial for comprehending the distinct behavioral traits of sophisticated network threats and for strengthening network security protocols.

iii. Methodology for Semi-supervised Learning

Ashfaq et al., [24] suggest that semi-supervised learning integrates both labeled and unlabeled samples to enhance the classifier's performance. Moreover, [25] argues that a semi-supervised Machine Learning approach use a pre-annotated dataset to imitate common patterns of behaviour. Semi-supervised learning combines the strengths of supervised and unsupervised learning techniques to create a model capable of classifying new data points in a dataset.

However, [25] suggested a two-stage semi-supervised statistical method for identifying network anomalies. The technique constructs a probabilistic model by utilizing pre-classified normal instances. Afterwards, this model is used to evaluate departures from the normal behaviour by using a predetermined threshold. The second phase involves implementing an iterative approach to reduce the frequency of false alarms. This is accomplished by employing a similarity distance and dispersion rate that are derived from the initial classifications of the probabilistic model [25].

Although the strategy has achieved high detection rates and low false positive rates, beating the Naïve Bayes algorithm in both true positive and false positive rates, it is still limited by the constraints of the anomaly detection method outlined in [26].

The author in [27] proposed a semi-supervised learning approach to address co-resident attacks in cloud-based systems. The approach incorporated a safeguarding mechanism that substantially enhances the computational expense required for a co-resident attack to achieve success in a virtual machine within a cloud computing system. The problem was framed as a 2-player security game, in which users were classified through the utilization of clustering analysis and semi-supervised Support Vector Machines (SVMs). Users are classified into three categories: high risk (malicious), medium risk (uncertain), and low risk (legal), depending on the modifications made to the virtual machine allocation method. This contributes to raising the overall expense for an attacker to execute a computationally demanding attack action, hence strengthening the defensive mechanism.

The technique successfully mitigated co-residence attacks by increasing the attacker's overall cost by a factor of 100. However, employing a single datacenter to execute the method is not feasible due to the requirement of addressing diverse scenarios across several datacenters, which may involve co-location and co-resident attacks.

The paper [28] examines cyber-attacks in computer networks using semi-supervised approach, offering an approach that integrates honeypots and network traffic manipulation to identify and prevent attacks proactively by anticipating the attacker's subsequent activities. Honeypots are recognized as essential instruments for assessing attacks, owing to the diverse range of implementations, resources, and insights derived from previous encounters. They are very effective at detecting and analyzing zero-day assaults. Honeypots are intentionally designed to monitor network traffic, regarding any incoming packets as potentially hostile, as any entity interacting with the honeypot is likely an attacker.

This method, however, has difficulties, particularly with faked traffic, as evidenced by major DDoS assaults on the Czech Republic's internet infrastructure. In this instance, both legitimate sites and honeypots inside the network were utilized as

reflectors, resulting in false positives that erroneously identified authentic targets as sources of malicious traffic. Notwithstanding the deceptive character of this traffic, it nonetheless contributed to the overall attack pattern.

A flow-based monitoring tool named Honeyscan was created, implemented, and evaluated on a live network to assess honeypot traffic. Acknowledging that not all phases of an assault are identifiable or transpire within the network, an anti-phishing framework, PhiGARO, was instituted for early detection in application-specific contexts. Moreover, prolonged network flow monitoring improved the identification of network spying activities. By examining application-level data instead of depending exclusively on the conventional 5-tuple flow record, the system attained enhanced detection accuracy.

The integration of an HTTP parser enhanced large-scale monitoring, allowing the system to trace HTTP requests aimed at the monitored network. This indicated that application-level scanning, especially via repeated HTTP queries, was a notable evidence of reconnaissance activities. By concentrating on the pattern of HTTP requests instead of the parameters utilized in each scan, the system effectively discovered scanning activity that may have otherwise remained unnoticed, especially those aimed at certain web apps.

Multi-stage assaults, encompassing both malevolent and innocuous stages, underscore the difficulty of detecting novel, intricate threats. The authors in [29] proposed a complete method employing a substantially Boosted Neural Network model to effectively detect multi-stage attack situations. The model exhibited significant predictive accuracy: 94.09% for the Quest model, 97.29% for the Bayesian Network, and 99.09% for the Neural Network. Upon assessment with the Multi-Step Cyber-Attack Dataset (MSCAD), the suggested Extremely Boosted Neural Network attained an impressive accuracy of 99.72% in forecasting multi-stage cyber-attacks.

The proposed model was constructed utilizing distinct Machine Learning methods in Python, implementing the QUEST, Bayesian Network, and Neural Network models in the preliminary phase to forecast multi-stage cyber-attacks in a cloud context. The model's performance was evaluated against multiple attack types, including Brute Force, HTTP DDoS, ICMP Flood, Normal traffic, Port Scanning, and Web Crawling. This thorough methodology highlights the model's capability to anticipate and mitigate a wide range of cyber-attacks with significant precision.

Industrial Control Systems (ICS) require extensive and efficient protection, particularly when they support vital infrastructure. Vasilomnolakis et al. [30] presented a novel honeypot aimed at identifying multi-stage assaults specifically directed at ICS networks. Upon the identification of a multi-stage attack, this honeypot is capable of generating attack signatures, enabling misuse-based Intrusion Detection Systems (IDSs) to thwart analogous attempts utilizing the

HOSTaGe honeypot. Given that honeypots have no other function, any engagement with them is intrinsically deemed suspicious, leading to an absence of false positives. A fundamental characteristic of honeypots is their capacity to remain undetected, masquerading as authentic gadgets instead than just decoys.

Shodan, an internet-connected device search engine, has devised techniques to identify honeypots by doing a series of probes and assessments, finally providing a score to each examined device. Shodan can ascertain the likelihood of a system being a honeypot based on this score, which may diminish its efficacy, as malware operators can leverage this information to evade recognized honeypots. The research findings indicate that the honeypot and its produced signatures attain a high level of detection accuracy. Furthermore, these signatures can be incorporated into the Bro IDS, allowing it to effectively thwart future assaults. The honeypot methodology is feasible for practical implementation and is compatible with current IDS frameworks. Shodan's identifying capabilities may diminish the usefulness of honeypots, as malware could circumvent well-known honeypots to avoid detection.

The authors in [31] proposed a comprehensive framework integrating anomaly and signature-based methodologies was established to proficiently detect both established and novel cyber threats. The framework analyzes incoming data packets using a Stacked Autoencoder, categorizing them as benign or malicious. The Grey Wolf Optimization technique extracts the most significant characteristics from packets designated as harmful. The system was evaluated using two prominent datasets, UNSW-NB15 and CIC-IDS-2017, attaining notable accuracy rates of 90.94% and 99.67%, respectively. This dual methodology, integrating statistical methodologies and deep learning techniques, exhibits robust proficiency in threat identification and classification.

Nonetheless, the signature-based element of the architecture has limits. Although it is proficient at addressing known threats, it encounters difficulties with novel or unrecognized attacks. Moreover, updating the signature database to incorporate the most recent attack definitions is a complicated and time-consuming endeavor, which may impede real-time threat detection for new cyber threats.

Hachimi et al. [32] developed a multi-stage Machine Learning-based intrusion detection system (ML-IDS) designed for 5G Cloud Radio Access Networks (C-RAN), focusing on the detection and classification of four types of jamming attacks: constant jamming, random jamming, deceptive jamming, and reactive jamming. The Wireless Sensor Network Dataset (WSN-DS), designed for wireless intrusion detection, was utilized to assess the system's efficacy. The multi-stage detection methodology, incorporating both supervised and deep learning classifiers, aims to decrease undiscovered attacks while reducing false negatives and false positives, attaining a detection and classification accuracy of up to 94%. Notwithstanding its superior performance, this solution possesses limitations. It is

incapable of detecting certain advanced jamming tactics, including shot noise-based intelligent jamming. The system also fails to address various other attack vectors aimed at C-RAN architecture, such as eavesdropping, primary user emulation, and impersonation attacks, which are outside its detection capabilities.

Multi-stage attacks can evolve considerably, resulting in considerable losses and damages for businesses. This research [33] introduces a framework that forecasts multi-stage assaults through a hybrid methodology, combining IP information evaluation and a process query system (PQS). This method is deemed useful for detecting multi-stage attacks; nevertheless, it necessitates prior knowledge of attack patterns (sequences), which can be difficult, as recognizing novel, intricate attacks frequently require time.

The identity checker, which is based on IP information, was assessed independently through a metrics-based methodology, resulting in excellent performance with no false positives and a high detection rate. Nonetheless, it is incapable of identifying multi-stage assaults if the associated IP addresses are not classified as malicious. The PQS technique is augmented by including additional models into its components, which are subsequently assessed independently using a metrics-based evaluation. To enhance overall system performance, analogous models will be amalgamated, minimizing the total number of models where feasible.

iii. **Reinforcement Learning-Based Approach**

Reinforcement learning is a form of Machine Learning that allows a software agent, such a sensor node, to gain information by actively interacting with its environment. The author in [34] stated that reinforcement learning is essential for pattern recognition since it allows software agents to learn from their interactions with the environment and make optimum decisions that maximize long-term rewards. Moreover, according to [35], reinforcement learning agents communicate within an initially unfamiliar environment and use the gained information to modify their action methods in order to maximize their rewards.

The discourse in [35] centers on reinforcement algorithms. The authors suggest that reinforcement learning is very suitable for addressing sequential problems, which may be described as Markov decision processes (MDPs), and therefore suitable for comprehending learning control problems. Supervised learning systems frequently encounter difficulties in comprehending these situations due to their intrinsic intricacy

The author in [36] employed fuzzy Q-learning to detect and prevent intrusions in Wireless Sensor Networks (WSNs). The concept utilizes a combination of cooperative game theory and fuzzy Q-learning algorithms to detect DDoS attacks. The methodology emulates sinkholes, a central station, and a malicious entity in a 3-player strategic game, wherein the game is initiated upon the transmission of a substantial volume of packets towards a certain node. Presently, the packets

received in the Wireless Sensor Network (WSN) are evaluated against a pre-established threshold to detect alarm occurrences. Once the threshold is surpassed, the technique initiates collaborative defense tactics to safeguard the sinkhole and the base station.

The NS-2 simulator was used to undertake a performance evaluation of the low energy adaptive clustering hierarchy (LEACH) technique through simulation. The purpose of this simulation was to demonstrate the precision of the technique in identifying and safeguarding against potential dangers. The method's architecture allows for the sink hole and base station to adaptively modify their approach in order to effectively detect and respond to an abrupt assault. The Intrusion Detection and Prevention System (IDPS) uses fuzzy Q-learning to continuously update its learning parameters in order to detect and prevent future attacks. This technique enables continuous learning from past attack patterns. Focusing solely on DDoS flooding attacks may make it challenging to determine its efficacy against other types of attacks. Hence, the model requires a thorough upgrade to strengthen its decision-making capabilities, specifically in identifying and mitigating novel forms of attacks.

Table1: Comparison of multi-stage attack detection methods.

TITLE [REF]	DATA SET	METHODOLOGY/ TOOLS	CONTRIBUTION	RESEARCH GAP
MARS:Multi-stage attackrecognition system[37,38]	LLS DDOS Data set	Supervised Approach/misuse	Multi-stageattackdetectionbasedon correlationofknowledge-basedand statisticalmodel	Vulnerabletozero-dayattackdueto dependencyonpre-defineddetails
Applications of hidden Markov models to detecting multi-stage network attacks [39]	Self generated dataset	Supervised approach	ApplyingHMMtomulti-stageattack detectionshowingthebestperformance amongC4.5,*k*-NNandHMM	Vulnerabletozero-dayattackdueto
A multi-stage attack mitigation mechanism for software-defined home networks [40]	Private data set	Supervised approach	ApplyingSDN/NFVenvironmentdatatoVulnerabletozero-multi-stageattackdetection	dayattackdueto dependencyonpre-defineddetails Experimentwithprivatedataonly
Multi-stage Jamming Attacks Detection using Deep Learning combined with kernelized support vector machine in 5G Cloud Radio Access Network (CRAN) [32]	Wireless sensor network dataset (WSN-DS)	Developing a new Machine Learning intrusion detection system (ML-IDS) based on supervised and deep learning classifiers model.	The system reduces the number of attacks missed, decrease the system's false negative and false positive rate. High detection and classification accuracy up to 94% as compared to only Multilayer Perceptron (MLP)	It only detects four types of network jamming attacks (constant, random, deceptive and reactive) It fails to detect other types of jamming attacks like shot noise-based Intelligent jamming. It fails to detect zero-day attacks and other types of Cloud Radio Access Network (CRAN) such as eavesdropping, primary user emulation and social engineering attacks

Early detection and mitigation of multi-stage network attacks [28]	Not stated	Uses honeypots and network traffic manipulation approach.	Flow-based monitoring of honeypot Early attack detection in application specific domain. Application-level network scanning detections Can handle zero-day attacks	Not effective in a spoofed network traffic. Some application level scans attacks such as repeating HTTPS request, would avoid detection Varying accuracy and inconsistent against different attacks. High computational requirements and false positive rates
Extremely boosted neural network for more accurate multi-stage cyber-attack prediction in cloud computing environment [29]	MSCAD Dataset	Extremely boosted neural network model	The model achieves 99.72% accuracy as compared to Quest model with 94.09%, Bayesian network with 97.29% and neural with 99.09%	Varying accuracy with limited test data The model is limited to Brute force, HTTP, DDOS, ICMP-flood normal, port scan, web-crawling attacks. Vulnerable to some zero-day attacks
Multi-stage intrusion detection system aided by grey wolf optimization algorithm[31]	UNSSW-NB15 and CIC-IDS-1017	It utilizes both signature based and anomaly-based techniques with grey wolf optimization algorithm	It can detect both unknown and known attacks with high accuracy	Inconsistency against different attacks with varying accuracy. High computational complexity according to time Vulnerable to zero-day attacks
Unsupervised multi-stage attack detection framework without details on single-stage attack[23]	DARPA LLS DDOS and CTU-13	Unsupervised approach	It can detect known and zero-day attacks without knowing pre-defined details on single-stage attack activities. Low false positive rate	Increasing computational complexity according to time Low understandability of multi-stage attack behavior.
Multi-stage attack detection and signature generation with ICS honeypots [30]	BRO-IDS	HOSTaGe honeypot development for attack detection and signatures generation for Bro IDS	The system supports existing IDS infrastructure. It can detect zero-day attacks. Not detected by Shodan search engine	The system becomes ineffective if the attacker avoids honeypot High computational complexity

Multi-stage attack detection via kill chain state machines [18]	CSE-CIC-IDS 2018	Uses a kill chain state machine that operate on clustered alert data to identify states and transitions of multi-stage attacks. Supervised approach	Substantially reduces the false positive alert correlation and attack contextualization It can be applied to any network-based alerts. It can detect complex attacks such as advance persistent threats (APT)	The algorithm only processes network level information and cannot handle host-level and user identity context attacks. Vulnerable to zero-day attack
A casual network-based system for predicting multi-stage attack with malicious IP [16]	DARPA'S Grand challenge problem GCP	Uses a probabilistic inference system for predicting multi-stage attack with malicious IP based on Bayesian belief network (BBN) model	Good in multi-stage attack prediction, detection and blacklisting of malicious IP addressing and computer network security in general.	It cannot predict attacks on devices that utilizes VPN. It cannot predict attacks using MAC address. Vulnerable to zero-day attack.
Multi-stage attack detection using contextual information[22]	Not stated	It is based on IDS exploits of contextual information in the form of point of life model and information related to expert judgment on the network behaviour by the use on FUZZY cognitive map(FCM)	It is able to efficiently detect the presence of a multi-stage attack in real time without prior training process. Can detect zero-day attack	The design of an FCM is very context-specific and may not easily generalized. Increased computational complexity
A novel multi-stage approach for hierarchical intrusion detection [17]	CIC-IDS 2017 and CSE-CIC-IDS 2018	Hierarchical intrusion detection approach is proposed	High adaptability without the necessity to retrain any of the classifiers. It can detect zero-day attack Hierarchical deployment ensures that privacy is preserved during training and operational service phases.	Cannot handle network level attacks High false positive alerts Varying accuracy and inconsistency against different attacks

4. Challenges in identifying multi-stage assaults

Identifying multi-stage attacks has several challenges. One of the challenges we face is dealing with the diverse problems related to wide-ranging intrusion detection. An obstacle that arises is the immense intricacy of contemporary network data, rendering the identification of pertinent security information arduous. In addition, the most dangerous attacks occur seldom, leading to a small number of attack occurrences in each dataset. Ourston et al. [39] offer additional elucidation on the Rare Data Problem, a difficulty encountered in intrusion detection research, along with other statistical concerns pertaining to security data, such as skewed distributions or imbalanced data sets.

The gravity of these challenges is intensified when considering multi-stage attacks. For a single-step attack, the trace often includes all the relevant information and is linked to a vulnerability in a system. The attack can be examined independently and contrasted with prior occurrences. As the number of processes engaged rises, it becomes more difficult to study and characterize the similarity between attacks. When conducting a multi-stage attack, it is essential to determine the specific attributes of each individual step as well as the correlation between them. While we may possess the ability to identify the steps involved, we may nevertheless fail to notice the attacking strategy.

Here, we will present a brief summary of some challenges faced while identifying multi-stage attacks.

i. The individual stages of a multi-stage attack may seem innocuous.

ii. An attacker can develop multiple strategies to carry out an attack [41]. Additionally, the execution of a plan may come to an end if the attacker loses interest or is incapable of taking use of the vulnerabilities in the network [42]. Also the interval between consecutive episodes of an assault can vary significantly, spanning from hours to days or even months [43].

iii. Technical limitations of network devices or their deployment or design may lead to the inability to recognize some processes, as mentioned in reference [43].

iv. An attacker is not required to follow a precise order when executing a multi-step assault [44], so the possible sequences of actions might be extremely complex.

v. Intrusion detection systems (IDS) often lack comprehensive information about the root causes of a problem [45, 46], making it difficult to identify contextual details.

vi. The majority of the features observed in traces are categorical, indicating that there is no inherent order or correlation among the possible values. This hinders the application of mathematical methods for creating attack scenarios [46].

vii. There is a lack of standardized datasets that may be used to evaluate the effectiveness of multi-stage attack detection systems. Furthermore, public research is limited in its access to a substantial fraction of the methodology and datasets employed by other researchers or commercial enterprises.

viii. A considerable proportion of the analysed methods for detecting multi-stage attacks depend on Intrusion Detection System (IDS) alerts as their main source of data. The production of IDS alerts is connected with several difficulties, including [47, 48, 49].

ix. Authentic notifications are frequently mixed with false positives and irrelevant ones.

5. Strategies for Cyber-Attack Prevention

The prevention of assaults is a proactive measure that swiftly discovers and addresses possible risks within a network. Prevention is highly pertinent in the process of mitigating cyber assaults. Most detection methodologies are reactive and are implemented only after significant damage has occurred in the affected area. Numerous intrusion prevention systems (IPSs) have been suggested to enhance cybersecurity.

Patil and Meshram [49] examined a strategy for thwarting network intrusions, focusing on varieties of Denial of Service (DoS) attacks, including flooding, IP spoofing floods, and ping of death attacks. The approach is designed to be platform-independent by utilizing the Java Virtual Machine (JVM). The packet sniffer is constructed with the Jpcap library, while the mitigation of malicious traffic invading the internal network is accomplished through the Linux iptables command. The utilization of the Jpcap library necessitates the prior installation of the WinPcaplibrary for Windows and the libpcap library for Unix or Linux systems. In this instance, attack prevention is executed by analysing inbound packets intercepted with Jpcap in promiscuous mode. When packets are analysed and the SYN flag is activated, consistently targeting the same destination address amongst ongoing network traffic, the system infers a SYN flood assault. The identified attack data is recorded in the log file, and subsequent measures are implemented to discard the packet using the iptables command (Linux) or net-filter (Windows). The suggested approach can also avoid smurf attacks (ICMP packets), SYN-FIN attacks, XMAS attacks, fraggle attacks (UDP packets), and all flag assaults. The method provided an expedited procedure for attack mitigation, albeit primarily reliant on iptables regulations. Furthermore, if an attack is not identified promptly by an expedited rule-matching procedure, it may penetrate the network and inflict significant harm to internal resources.

One of the most elusive forms of cyber-attack is the Distributed Denial of Service (DDoS) flooding attack, which employs botnets to obstruct services intended for legitimate users of a system or network. Botnets are frequently utilized to inundate several computers, perhaps on a global scale, with malicious packets via the Internet by exploiting weaknesses in these systems.
These botnets consist of elements referred to as masters, handlers, and bots. The attackers are the orchestrators, who interact with the operatives or bots through intermediaries.

The attackers utilize compromised systems for command-and-control operations. Zombies are compromised computers that constitute an attack force, systematically infiltrating other vulnerable systems along the attack vector to amplify the assault until all computing resources are rendered inoperative, potentially resulting in significant and irreversible damage.

Countering DDoS attacks has proven to be an arduous endeavour. Ideally, numerous computers connected to Internet-enabled networks, together with devices such as smartphones and personal digital assistants (PDAs), should be safeguarded against all types of vulnerable services and ports. Nonetheless, these susceptible services and ports result from unpatched systems, for which the majority of security updates transmitted to devices via proxy servers are neither implemented nor timely executed. This results in numerous unpatched and insecure devices that can be exploited in a DDoS flooding assault utilizing the command and control (C & C) functionality of botnets.

Therefore, [50] asserts that techniques such as source address authentication, capabilities, and filtering are crucial for mitigating DDoS flooding attacks. Internet service providers should collaborate by utilizing technologies such as cloud computing and IoT to prevent and mitigate threats, leveraging the closeness of the Internet as a significant advantage.

Moreover, while the classification provided has taken into account various sources and results of DDoS assaults, little focus has been placed on the correlation of warnings, which could potentially identify the attack's source when a significant number of cases are present. When evaluating flooding assaults based just on the volume of traffic produced, without examining the causal linkages among traffic occurrences, identifying the source of the attack in real time might be challenging. A software-defined intrusion prevention system, referred to as SDNIPS, is proposed for mitigating cloud-based assaults in [51]. The methodology incorporates a detection element that integrates a Snort-based intrusion detection system with Open vSwitch (OVS). The architecture of SDNIPS is further streamlined, enabling cloud resources to create traffic that traverses the SDNIPS agent. The traffic is subsequently compared to the Snort rules, and any matches are flagged as alerts that enhance the log file. The SDNIPS daemon captures this warning information and transmits it to the JSON server at the controller's end. During the processing of alert information, the alert interpreter analyses the data and extracts essential details, such the attack type, source and destination IP addresses, and TCP port, among others.

During the further processing of alert information, the OVS modifies the flow table utilizing the OpenFlow rule entries from the rule's generator. Any dubious traffic corresponding to the revised flow table entries is subsequently addressed by implementing the requisite countermeasures in the data plane. This method truncates an attack, so safeguarding cloud resources from compromise. The methodology exhibits an effective preventative system; nevertheless, the reliance on Snort necessitates complete dependency on specialist knowledge for rule definition, which might be time-consuming. This strategy will likely result in increased use of computing resources, as traffic must continuously traverse the IPS, and delays in matching each traffic pattern to the established rules will considerably affect the host system's resources.

As organizational security requirements escalate, increasingly sophisticated solutions are necessary to safeguard extensive resource allocations. A crucial element in attaining a viable security solution is the availability of a cost-effective, flexible, and scalable product or methodology. This is the emphasis of [52] in their suggested real-time methodology for identifying and mitigating assaults. The methodology, grounded on the software engineering framework such as requirement analysis, design, implementation, and testingconfigures Snort in inline mode to facilitate intrusion prevention. Configuring Snort in inline mode enables the Intrusion Prevention System (IPS) to position its sensors for the interception and elimination of suspicious packets that are likely to contain attack payloads. The discarded packets are ultimately recorded in Splunk. Despite this, the incapacity of signature-based intrusion detection and prevention systems, such as Snort, to identify unknown attacks, along with its inadequate performance under excessive network traffic, constitutes a significant limitation of this method [53].

6. Evaluations, obstacles, and future prospects

Table 1 presents a comparison of various multi-stage attacksdetection strategies. The comparison is based on the methodology/tool used, the training/testing data set employed, as well as the contributions and research gaps identified. Out of the thirteen detection strategies analysed in the table, nine of them utilize a combined total of nine publicly accessible data sets for both training and testing the model. One scheme utilizes a private dataset, with one plan employing a self-generated data set, while for the remaining schemes, the specific data sets are not specified. Furthermore, these nine public data sets originate from various domains such as malware and network attacks, and they vary in terms of their feature space. The evaluated strategies demonstrate potential in effectively identifying multi-stage attacks through various methodologies. Nevertheless, the majority of detection techniques frequently demonstrate significant fluctuations in their accuracy when faced with various types of attacks. Hence, it is challenging to quantitatively compare various schemes due to the differences in evaluation data sets, assessment measures, and comparison schemes. The available training/testing data sets are constrained and isolated, and there is a notable absence of a comprehensive, representative data collection at a significant scale. Various methods are commonly assessed using distinct sets of performance indicators. Because implementing comparison schemes requires a substantial amount of effort, the examined schemes are typically compared to only a few rival systems before drawing conclusions. The outcomes are often inconsistent and difficult to interpret due to the use of non-representative data sets and a restricted number of model comparisons.

The multi-stage detection systems listed in Table 1 can be classified into four categories: Unsupervised, semi-supervised, supervised and reinforcement Learning methodologies.

Unsupervised learning refers to a category of Machine Learning algorithms that extract patterns from data that lacks explicit labelling. The objective of the assault's detection model is to acquire knowledge about a condensed representation of regular data in order to identify attacks. Supervised and semi-supervised attack detection approaches employ either supervised learning or a combination of supervised and unsupervised learning techniques. Supervised and hybrid attack detectors can provide precise detection by utilizing representative training data sets. Regrettably, the data collection does not include any instances of certain types of assaults, such as zero-day attacks. The detection techniques must make the assumption that zero-day attacks exhibit similar behaviour to known attacks, a hypothesis that has not yet been verified.

The speed of training and detection is a crucial component in multi-stage attack detection. Although training often requires more time than detection, the examined approaches are all capable of completing both training and detection within a suitable timeframe, despite variations in pace across different methods.

6.1 Identified Research Gaps

Despite the significant advancements in machine learning-based approaches for multi-stage cyber-attack detection, several critical research gaps remain.

1. Lack of Zero-Day Attack Detection

One of the major limitations of existing models is their inability to effectively detect zero-day attacks. Most supervised and hybrid learning approaches rely heavily on labeled datasets containing known attack patterns, making them less effective against novel threats [16], [31]. As a result, these models struggle to identify previously unseen or evolving attack strategies, leaving systems vulnerable to emerging cyber threats.

2. Dataset Limitations

The performance of machine learning models is highly dependent on the quality and diversity of training datasets. However, many existing studies rely on limited or domain-specific datasets such as CIC-IDS and DARPA datasets [17], [23]. These datasets often fail to represent real-world network environments, which reduces the reliability and generalizability of detection models in practical deployments.

3. High Computational Complexity

Advanced machine learning and deep learning models often require significant computational resources for training and real-time detection. Techniques such as ensemble learning, deep neural networks, and optimization-based models introduce high processing overhead, limiting their applicability in real-time and resource-constrained environments [31], [32].

4. Lack of Generalization

Many proposed models demonstrate high accuracy when evaluated on specific datasets but fail to generalize across different network environments. This limitation arises due to overfitting and the dependency on specific training data distributions, which reduces their effectiveness in dynamic and heterogeneous network conditions [17], [22].

6.2. Prospects for the future

Additional endeavours are necessary to tackle the difficulties in formulating efficient multi-stage attack detection techniques. To resolve the problem of insufficient zero-day attack information in the training data set, one can utilize a honeypot [296] to gather the zero-day attack data prior to their discovery. Utilizing domain expertise to perform feature engineering is an additional method to enhance the accuracy of detection. Attackers can evade detection if the characteristics of their attack closely resemble those of lawful activities. Incorporating the expertise of domain specialists is crucial for effectively incorporating their knowledge into the process of feature engineering. This ensures that the newly developed attacks will be detectable within the specified feature space.

Detection techniques are advancing rapidly. Utilizing the most recent breakthroughs in Machine Learning and implementing the latest Machine Learning models is an additional method to protect against certain types of attacks, such as zero-day attacks. Reinforcement Learning (RL) is a form of Machine Learning where an agent, or decision maker, learns in an interactive environment through trial and error, using feedback from its own actions and experiences. The agent must only have access to induced feedbacks, without the necessity to possess knowledge of all the components that determine these feedbacks. Reinforcement Learning (RL) is especially suitable for multi-stage cyber assault challenges that involve unknown vulnerabilities and attack targets.

The creation of a multi-stage attacksdetection benchmark will help overcome many obstacles that impede research and development progress. An extensive benchmark suite with standardized data sets, a wide range of representative models, and automated testing and assessment capabilities will significantly accelerate the development of multi-stage attack detection systems.

7. Summary

Multi-stage attacks frequently occur and result in significant financial losses, as well as potential damage to the reputation of both organizations and individuals. Machine Learning-based detection is the most promising and successful approach for detecting multi-stage attacks.

This paper presents a thorough examination of multi-stage attack detection methods. The review focuses on the different types of Machine Learning methods used, such as supervised, un-supervised, semi-supervised and reinforcement

approaches. Figure 1 provides a visual representation of these methods. Nevertheless, the Machine Learning-based detection method encounters a fundamental difficulty in that it does not have the ability to represent zero-day attacks in the datasets. The restricted and isolated datasets, along with the incomplete range of features, significantly diminish the accuracy, resilience, and dependability of the models, falling short of the necessary level. In order to make progress, we suggest utilizing the most recent developments in Machine Learning research and integrating the expertise of domain specialists more effectively into the building of the Machine Learning model. Furthermore, the creation of a comprehensive and standardized benchmark that contains abundant data will greatly aid in the ongoing enhancement of multi-stage attack detection models. Additionally, addressing the research gaps mentioned is essential for developing more robust, scalable, and adaptive multi-stage attack detection systems. Future research should focus on designing models that can generalize across environments, efficiently detect zero-day attacks, and operate with reduced computational overhead while leveraging more realistic and comprehensive datasets.

REFERENCES

[1] A. L. Buczak and E. Guven, "A survey of data mining and Machine Learning methods for cyber securityintrusion detection", IEEE Communications Surveys & Tutorials, vol. 18, no. 2, **(2016)**, pp. 1153-1176.

[2] U. Akyazi, "Possible scenarios and maneuvers for cyber operational area", In European Conference onCyber Warfare and Security, Academic Conferences International Limited, Greece, **(2014)** July 3-4.

[3] D. E. Denning, "Framework and principles for active cyber defense", Computers & Security, vol. 40,**(2014)**, pp. 108-113.

[4] G. Kim, S. Lee and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection", Expert Systems with Applications, vol. 41, no. 4, **(2014)**, pp. 1690-1700.International Journal of Security and Its Applications Vol. 12, No. 4 (2018)28 Copyright ? **2018** SERSC Australia

[5] S. Yoo, S. Kim, A. Choudhary, O. P. Roy and T. Tuithung, "Two-phase malicious web page detection scheme using misuse and anomaly detection", International Journal of Reliable Information and Assurance, vol. 2, no. 1, **(2014)**, pp. 1-9.

[6] M. S. Rani and S. B. Xavier, "A Hybrid Intrusion Detection System Based on C5. 0 Decision Tree Algorithm and One-Class SVM with CFA", International Journal of Innovative Research in Computer, vol. 3, no. 6, **(2015)**, pp. 5526-5537.

[7] H.Sugumaran, M., and Balasaraswathi, V. R. (2016). Ids using Machine Learning-current state of art and future directions. *British Journal of Applied Science and Technology, 15*(3).

[8] M. Jacob, and Wanjala, M. Y. (2018). A Review of Intrusion Detection Systems. *Global Journal of Computer Science and Technology.*

[9] S. Biswas (2018). Intrusion detection using Machine Learning: A comparison study. *International Journal of pure and applied mathematics, 118*(19), 101-114.

[10] A. Choudhury and D. Gupta (2019). A survey on medical diagnosis of diabetes using Machine Learning techniques. In *Recent developments in Machine Learning and data analytics* (pp. 67-78). Springer, Singapore.

[11] A. Ghosh, E. Fassnacht, Joshi, P. K., Koch, B., 2014. A framework for mapping tree species combining hyperspectral and LiDAR data: role of selected classifiers and sensor across three spatial scales. Int. J. Appl. Earth Obs. Geoinf. 26, 49-63.

[12] S. Kiran, Devisetty, R. K., Kalyan, N. P., Mukundini, K., and Karthi, R. (2020). Building an intrusion detection system for iot environment using Machine Learning techniques. *Procedia computer science, 171, 2372-2379*

[13] X. Liu, Yang, Y., Choo, K. K. R., and Wang, H. (2018). Security and privacy challenges for internet-of-things and fog computing.

[14] M. Chen, Wan, J., and Li, F. (2012). Machine-to-machine Communications:

Architectures, Standards and Applications. *KSII Transactions on Internet and Information Systems, 6.* https://doi.org/10.3837/tiis.2012.02.002

[15] A. Tabassum, and Lebda, W. (2019). Security Framework for IoT Devices against Cyber- Attacks. arXiv preprint arXiv: 1912.01712.

[16] A. Osarumwense, E.Oghenerukevbe (2020) "A casual network-based system for predicting multi-stage attack with malicious IP". International journal of Academic multidisciplinary research (IJAMR), vol 4, issue 5, may, 2020. Page 1-8.

[17] V. Miel, L. D'hooge, T. Wauters, B. Volckaert and F. De turck "A novel Multi-stage approach for Hierarchical intrusion detection"

[18] W. Florian, F. Ortmann, S. Hass, M. Vallentin, M. Fischer (2021),' Multi-stage Attack Detection via kill chain state machine'.

[19] J. Song, H. Takakura, Y. Okabe and K. Nakao, "Toward a more practical unsupervised anomaly detection system", Information Sciences, vol. 231, **(2013)**, pp. 4-14.

[20] A. Abduvaliyev, A. S. K. Pathan, J. Zhou, R. Roman and W. C. Wong, "On the vital areas of intrusion detection systemsin wireless sensor networks", IEEE Communications Surveys&Tutorials, vol.15,no. 3, **(2013)**, pp. 1223-1237.

[21] I. Butun, S. D. Morgera and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks", IEEE communications surveys & tutorials, vol. 16, no. 1, **(2014)**, pp. 266-282.

[22] K. Kyriakopoulos, F.Aparicio-Navarro, I.Ghafir, S. Lambotharan, and J. Chambers. 2019. "Multi-stage Attack Detection Using Contextual Information". figshare. https://hdl.handle.net/2134/34219

[23] S. Jinmyeong, C. Seok-Hwan, P. Liu and C. Yoon-Ho (2019) 'Unsupervised multi-stage attack detection framework without details on single-stage attacks; Future generation computer systems 100(2019) 811-825

[24] R. A. R. Ashfaq, X. Z. Wang, J. Z. Huang, H. Abbas and Y. L. He, "Fuzziness based semi-supervised learning approach for intrusion detection system", Information Sciences, vol. 378, **(2017)**, pp. 484-497.

[25] N. B. Aissa and M. Guerroumi, "Semi-supervised statistical approach for network anomaly detection", Procedia Computer Science, vol. 83, **(2016)**, pp. 1090-1095.

[26] G. Kim, S. Lee and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection", Expert Systems with Applications, vol. 41, no. 4, **(2014)**, pp. 1690-1700.

[27] Y. Han, T. Alpcan, J. Chan, C. Leckie and B. I. Rubinstein, "A game theoretical approach to defend against co-resident attacks in cloud computing: Preventing co-residence using semi-supervised learning", IEEE Transactions on information Forensics and Security, vol. 11, no. 3, **(2016)**, pp. 556-570.

[28] H. Martin (2015) "Early detection and mitigation of multi-stage network attacks PhD thesis in Masarykovaunwerzitafukuita informality January, 2015

[29] D. Surjeet, P.Manoharan, L. Kumar, B. Seth, D.Alsekait, S.Simaiya, M.Hamdi, K.Raahemifar (2023) "Extremely boosted Neural network for more accurate multi-stage cyber-attack prediction in cloud computing environment." Journal

of cloud computing: Advanced, systems and applications, 2023.

[30] V. Emmanouil, S. Srinivasa, C. Garcia Cordero, M. Muhlhauser (2016) Multi-stage Attack Detection and Signature Generation with ICS Honeypots'

[31] C. Somnath, V. Shaw and R. Das (2021) , 'Multi-stage Intrusion DetectionSystem aided by grey wolf optimization algorithm.' Spriger nature 2021. http://doi.org/10.21203/rs.3.rs-2680915/v1

[32] H. Marouane, G. Kaddoum, G. Gagnon and P. Illy (2020) "multi-stage jamming attacks detection using deep learning combined with kernelized support vector machine in 5g cloud radio access network 2020 IEEE international symposium on networks computers and communications (ISNCC'20), October, 2020.

[33] Z. Abdulrazaq, J. flint, D. parish (2015)," predicting multi-stage Attacks based on hybrid approach," international journal for information security Research (IJISR), VOL 5, issue 3, September 2015, pp582-590.

[34] M. A. Alsheikh, S. Lin, D. Niyato and H. P. Tan, "Machine Learning in wireless sensor networks: Algorithms, strategies, and applications", IEEE Communications Surveys and Tutorials, vol. 16, no. 4, **(2014)**, pp. 1996-2018.

[35] X. Xu, L. Zuo and Z. Huang, "Reinforcement learning algorithms with function approximation: Recent advances and applications", Information Sciences, vol. 261, **(2014)**, pp. 1-31.

[36] S. Shamshirband, A. Patel, N. B. Anuar, M. L. M. Kiah and A. Abraham, "Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks", Engineering Applications of Artificial Intelligence, vol. 32, **(2014)**, pp. 228-241.

[37] F. Alserhani, M. Akhlaq, I.U. Awan, A.J. Cullen, P. Mirchandani, Mars: Multi-stage attack recognition system, in: 2010 24th IEEE International Conference on Advanced InformationNetworkingandApplications,2010,pp.753–759, http://dx.doi.org/10.1109/AINA.2010.57,

[38] F. Alserhani, A framework for multi-stage attack detection, in: 2013 SaudiInternationalElectronics,CommunicationsandPhotonicsConference,2013, pp.1–6, http://dx.doi.org/10.1109/SIECPC.2013.6550973.

[39] D.Ourston,S.Matzner,W.Stump,B.Hopkins,Applicationsofhiddenmarkov models to detecting multi-stage network attacks, in: 36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the, 2003, p.10, http://dx.doi.org/10.1109/HICSS.2003.1174909.

[40] S.Luo, J.Wu, J.Li, L.Guo, A multi-stage attack mitigation mechanism for software-defined home networks, IEEE Trans. Consum. Electron. 62 (2)(2016)200–207,http://dx.doi.org/10.1109/TCE.2016.7514720

[41] X. Qin, W. Lee, Attack plan recognition and prediction using causal networks, in:20th Annual Computer Security Applications Conference, IEEE, 2004, pp. 370–379. doi:10.1109/CSAC.2004.7

[42] S. J. Yang, J. Holsopple, M. Sudit, Evaluating threat assessment for multi-stage cyber-attacks, in: MILCOM 2006-2006 IEEE Military Communications

conference, IEEE, Washington, D.C., 2006, pp. 1-7.doi:10. 1109/MILCOM. 2006. 302216.

[43] B. Chen, J. Lee, A. S. Wu, Active event correlation in Bro IDS to de- tect multi-stage attacks, in: Fourth IEEE International Workshop on Information Assurance (IWIA'06), IEEE, 2006, pp. 16 pp.–50.doi:10.1109/IWIA.2006.2.

[44] Z. Zhang, P.-H. Ho, X. Lin, H. Shen, Janus: A two-sided analytical model for multi-stage coordinated attacks, in: 9th International Conference on Information Security and Cryptology, Springer, Busan, Korea, 2006, pp.136{154. doi:10.1007/11927587_13.

[45] S. Salah, G.Maci´a-Fern´andez, J.E.D´ıaz-Verdejo, A model-based survey of alert correlation techniques, Computer Networks 57 (5) (2013) 1289–1317.

[46] K. Julisch, Mining alarm clusters to improve alarm handling efficiency, in: ComputerSecurityApplicationsConference,2001.ACSAC2001. Proceedings 17th Annual, IEEE, 2001, pp. 12–21.

[47] P. Ning, D.Xu, Toward automated intrusion alert analysis, Springer, 2010, pp. 175–205.

[48] D.Xu, P.Ning, Alert correlation through triggering events and common resources, in: Computer Security Applications Conference, 2004. 20th Annual, IEEE, 2004, pp. 360–369.

[49] S. Patil and B. B. Meshram, “Intrusion Prevention System”, International Journal of Emerging trends in Engineering and Development, vol. 4, no. 2, **(2012)**, pp. 577-584.

[50] S. T. Zargar, J. Joshi and D. Tipper, “A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks”, IEEE Communications Surveys & Tutorials, vol. 15, no. 4, **(2013)**, pp.2046-2069.

[51] T. Xing, Z. Xiong, D. Huang and D. Medhi, “SDNIPS: Enabling Software-Defined Networking based intrusion prevention system in clouds”, In Network and Service Management (CNSM), 10thInternational Conference, IEEE, **(2014)**, pp. 308-311.

[52] P. S. Kenkre, A. Pai and L. Colaco, “Real time intrusion detection and prevention system”, In Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA), Springer, Cham, **(2014)**, pp. 405-411

[53] A. Abduvaliyev, A. S. K. Pathan, J. Zhou, R. Roman and W. C. Wong, “On the vital areas of intrusion detection systems in wireless sensor networks”, IEEE Communications Surveys & Tutorials, vol. 15, no.3, **(2013)**, pp. 1223-1237.

A HYBRID MACHINE LEARNING FRAMEWORK FOR SOFTWARE DEFECT PREDICTIONUSING NASA MDP DATASETS.

NWACHUKWU-NWOKEAFOR, K. C.
Michael Okpara University of Agriculture, Umudike,
nwachukwu.nkenneth@mouau.edu.ng, nwachukwuken72@gmail.com.

ABSTRACT

This research presents the development of AI-based Hybrid Model for software testing and bug prediction system leveraging supervised machine learning models and advanced feature engineering techniques on the publicly available NASA Metrics Data Program (MDP) datasets. Specifically, this study presented a modular and generalizable machine learning framework that integrates multiple preprocessing techniques that include Synthetic Minority Over-sampling Technique (SMOTE), Adaptive Synthetic Sampling (ADASYN), and three distinct feature selection methods: Mutual Information (MI), Sequential Feature Selection (SFS), and the Boruta algorithm. The framework is implemented and validated across twelve NASA MDP datasets (e.g., CM1, JM1, KC1, MC1), ensuring a broad assessment of model robustness and generalizability across diverse software environments. Four classification models; Logistic Regression, Support Vector Classifier (SVC), Random Forest, and XGBoost were evaluated and compared to benchmark the prediction performance. These models were assessed using evaluation metrics that include Precision, Recall, F1-Score, Area Under the Curve (AUC), and Matthews Correlation Coefficient (MCC), with additional emphasis on addressing class imbalance and enhancing interpretability. The findings reveal that ensemble learning models, particularly XGBoost integrated with Boruta and SMOTE, consistently outperform baseline classifiers, achieving an average F1-Score of over 0.89 across most datasets while maintaining model interpretability and computational efficiency. In addition to performance benchmark, the study presents a comprehensive feature Hybrid importance analysis, identifying critical software metrics across models that contribute to early defect prediction. The research demonstrates not only the feasibility of implementing an automated. AI-based Hybrid Model for bug Testing and Prediction System in real-world settings but also contributes to a deeper understanding of software metrics and their role in predictive maintenance.

Keywords; Hybrid Model, Artificial Intelligence, Software Bug Testing and Prediction.

1. BACKGROUND

A crucial part of the software development lifecycle, software quality assurance (SQA) seeks to guarantee that software products are error-free and meet predetermined standards. As software systems get more complex, automated software defect prediction (SDP) techniques are being investigated because traditional testing methods are frequently insufficient to identify all possible

problems (Challagulla *et al.*, 2008; Malhotra, 2015). Modern software systems depend heavily on software quality because errors and flaws can result in serious financial losses, harm to one's reputation, and security risks. In the United States, software failures cost businesses more than $2 trillion a year, according to research by the Consortium for information technology (IT) software quality. Most faults are discovered after the product has been deployed (CISG, 2022). Traditional methods of bug discovery and testing are not working well in light of the quick evolution of software development paradigms like Agile and DevOps. These methods, which can be laborious and prone to human mistake, frequently include post-deployment testing, static analysis, and manual code reviews.Software defect prediction is the process of identifying faulty code components by applying statistical methods and machine learning (ML). The goal of these strategies is to optimize testing efforts by predicting which modules are likely to have defects by analyzing historical data, such as source code metrics, modification history, and defect logs (Hall *et al.*, 2012; Kamei *et al.*, 2013). Over the years, various models, including traditional machine learning algorithms (e.g., Naive Bayes, Decision Trees, SVM) and, more recently, deep learning models, have been proposed for SDP (Wang and Liu, 2021).Due to their interpretability and very low computing cost, traditional machine learning models have been widely used in defect prediction. By using software measures like; cyclomatic complexity, code churn, and lines of code, techniques like Logistic Regression, Random Forest, and Naive Bayes have shown a moderate level of success in finding problematic modules (Ghotra *et al.*, 2015). However, these models might not reflect the complex interactions within the code structure and frequently necessitate considerable feature engineering (Rahman and Devanbu, 2018). Since they can automatically acquire hierarchical features from raw data, deep learning models in particular, Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have been more and more popular for SDP in recent years (Zhang*et al.*, 2020). By using pre-trained language models to comprehend code semantics, transformer models such as BERT and CodeBERT have further transformed defect prediction (Feng *et al.*, 2020). By identifying both local and global dependencies in source code, transformers have demonstrated great promise in defect prediction, increasing the precision of defect detection.Although SDP has advanced, there are still a number of obstacles to overcome. While deep learning models necessitate significant processing resources and big labelled datasets, traditional models frequently suffer from the "curse of dimensionality" because of their high feature space (Arora and Sinha, 2019). Furthermore, developers may find it challenging to trust the predictions in the absence of explicit explanations because to the interpretability issues with deep learning models (Ghotra *et al.*, 2015).

Ø RESEARCH GAP

The purpose of this study is to explore how machine learning (ML) models may be used to proactively find sections of big codebases that are prone to bugs prior to code deployment. The work intends to create a prediction model that may foresee

software flaws early in the development process by utilizing previous data on code changes, bug reports, and software metrics. The main objective is to raise the general caliber of software products while lowering the time and expense involved in bug fixes.

2. MATERIALS AND METHODS

This section presents a comprehensive description of the materials utilized throughout the study. These include the datasets, computational tools, programming libraries, development platforms, and the feature selection algorithms employed. Each component played a crucial role in implementing and evaluating the proposed machine learning models for bug prediction.

Ø Dataset Description

The dataset used in this study is obtained from the NASA Metrics Data Program (MDP) repository, a well-known and publicly available source of labelled software defect datasets. The datasets contain historical software metrics derived from real-world NASA software projects, with each instance representing a software module described by various static code attributes. The target variable is binary in nature, denoting whether a software module is fault-prone or non-fault-prone. The datasets are characterized by:

i. A diverse range of metrics including size, complexity (e.g., McCabe metrics), and

Halstead metrics. (ii) Imbalanced class distribution, which is common in defect prediction datasets due to the naturally lower occurrence of defective modules compared to non-defective ones.

Ø Tools and Libraries

The study was implemented using Python programming language (version 3.13) due to its widespread adoption in data science and machine learning applications. A combination of scientific and machine learning libraries was used to perform pre-processing, modelling, and evaluation. These libraries include:

i. Pandas and Numpy: For data manipulation and numerical operations.

ii. Scikit-learn: For model building, evaluation, and implementation of feature selection techniques. (iii) XGBoost: For the implementation of gradient boosting models.

iii. BorutaPy: A wrapper for implementing the Boruta feature selection algorithm.

iv. Matplotlib and Seaborn: For visualization of comparative model performances and analysis

v. Imbalanced-learn: To address class imbalance via techniques such as SMOTE (if applied).

Ø Computational Environment

All experiments and model evaluations were executed in a local computing environment with the configurations:

i. Operating System: Windows 10 (64-bit)

ii. Processor: Intel® Core™ i5 @ 3.2GHz. (iii) RAM: 16GB DDR4

iii. Storage: 512GB HDD. (iv) Python Environment: Pycharm

This environment ensured optimal performance and reproducibility of results, allowing consistent evaluation across multiple experimental setups.

Ø System Model Developed.

The system model for bug prediction is a multi-phase pipeline that integrates pre-processing, feature selection, machine learning model training, and performance evaluation. The design leverages state-of-the-art algorithms to ensure data quality, dimensionality reduction, and accurate classification of software modules as fault-prone or non-fault-prone. The system model is modular, allowing for systematic experimentation and optimization at each stage. Figure1, illustrates the overall architectural flow of the system.

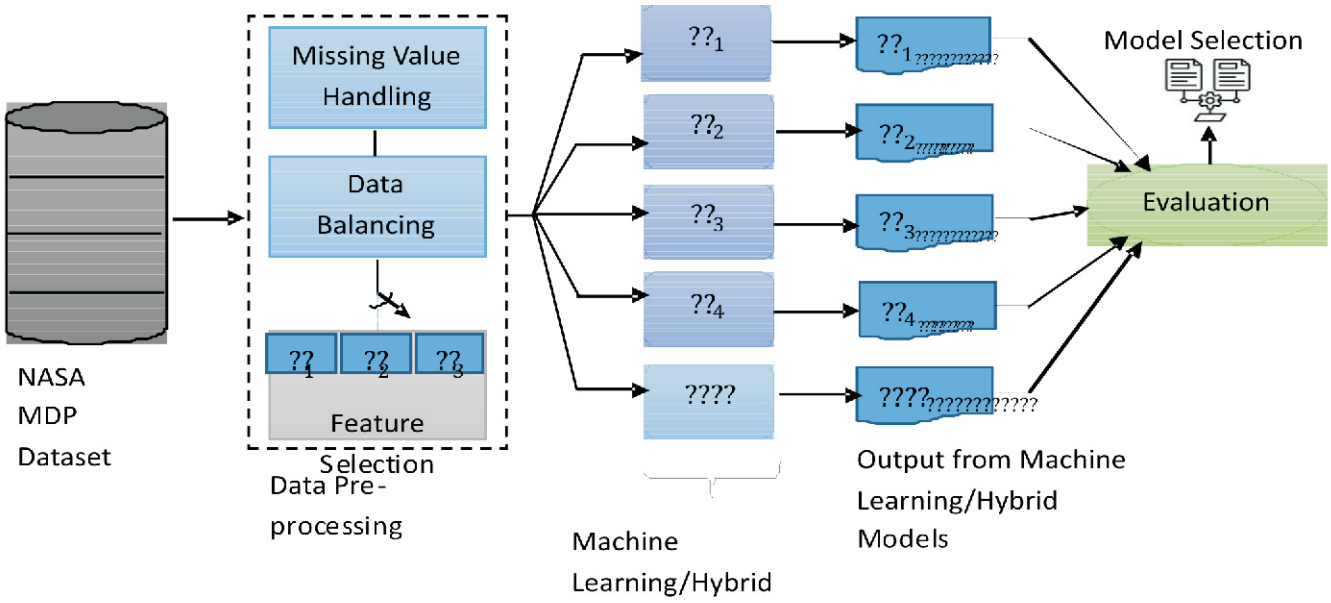


Figure 1: The proposed system model.Source: Researcher (2025)

Ø Overview of the System Model

The proposed system model comprises five major components, namely:

i. Data acquisition and pre-processing. (ii)Feature selection. (iii) Model training and classification. (iv)Model evaluation. (v) Model selection and interpretation

Ø Data Acquisition and Pre-processing

The system begins by acquiring labelled software metrics data from the NASA MDP repository, comprising both numerical and categorical features describing software modules. To ensure data quality and suitability for modelling, the following pre-processing techniques were systematically applied:

i. **K-Nearest Neighbors (KNN) Imputation:** Used to handle missing values in the dataset by estimating them based on the nearest neighbors. This technique preserves the data distribution more effectively than mean or median imputation.
ii. **Recursive Feature Elimination (RFE):** Employed as a preparatory feature reduction mechanism by recursively removing less important features based on model weights, helping to reduce overfitting and improve generalization.
iii. **SMOTE (Synthetic Minority Oversampling Technique):** Used to address class imbalance in the dataset by synthetically generating new instances of the minority class. This improves the classifier's ability to detect minority (fault-prone) modules without being biased toward the majority class at the end of this phase, a clean, balanced, and dimensionally reduced dataset is obtained, ready for feature selection and model training.

Ø Feature Selection

Following pre-processing, three prominent feature selection techniques are applied to identify the most informative attributes for bug prediction:

i. **Mutual Information (MI):** A filter-based method that quantifies the mutual dependency between each feature and the target variable. Features with high MI scores are retained for modelling.
ii. **Sequential Feature Selection (SFS):** A wrapper-based forward selection technique that incrementally builds an optimal feature subset based on model performance during cross-validation.
iii. **Boruta Algorithm:** A wrapper-based, all-relevant feature selection approach built on Random Forest, which compares actual features with randomized shadow features and retains only the statistically significant ones. These methods help reduce redundancy and irrelevance in the data, enhancing model interpretability and efficiency.

Ø Model Training and Classification

The processed and feature-selected datasets are subsequently used to train four supervised learning models:

(i) **Logistic Regression (LR):** A probabilistic linear classifier useful for binary classification tasks.
(ii) **Random Forest (RF):** An ensemble method using multiple decision trees with bagging to improve robustness and accuracy.
(iii) **Extreme Gradient Boosting (XGBoost):** An advanced boosting algorithm known for its speed and performance in structured data.
(iv) **Support Vector Classifier (SVC):** A kernel-based classifier that constructs optimal hyper planes for classification, particularly effective in high-dimensional

spaces.

(v) **Hybrid Model:** The best performing models are selected to form the hybrid model to complement each other and enhance the general model output. Each model is trained separately on the datasets obtained from the three feature selection methods, as well as on the original dataset without feature selection for baseline comparison.

Ø Model Evaluation and Selection

After training, the performance of each model is rigorously evaluated using the following metrics:

i. **Accuracy:** Measures the overall correctness of predictions. (ii) **Precision:** Evaluates the ability of the model to identify true positives among predicted positives. (iii) **Recall:** Assesses the model's capability to detect actual positives. (iv)**F1-Score:** Harmonic mean of precision and recall, providing a balanced measure. (v)**AUC (Area-Under the Curve):** Indicating the model's discriminative ability across different threshold settings.(vi)**Training Time:** Captures the computational efficiency of each modelling approach with respect to time.

Ø Implementation of Synthetic Minority Over-sampling Technique (SMOTE)

In supervised learning, especially in real-world software engineering datasets such as the NASA MDP dataset, data imbalance is a prevalent challenge. Class imbalance, where the majority of software modules are non-defective and only a minority are defective (bug-prone), causes bias in classification models toward the majority class. To improve model performance, the data must be pre-processed, and balanced using the Synthetic Minority Over-Sampling Technique (SMOTE) and Adaptive Synthetic Sampling (ADASYN) method respectively, and explored through various visual and statistical techniques. This process ensures better insight into feature distributions, relationships, and outliers, aiding both model interpretability and accuracy. Also, this technique is deployed to handle class imbalance (bugs vs. non-bugs) and explore key patterns and relationships in the dataset. After data balancing is applied, each feature is explored using histograms or density plots to understand its distribution. For example, Lines of Code (LOC) in modules tend to have a long tail, and visualizing this helps determine the need for normalization. Let $X = \{x_1, x_2, \ldots x_n\}$ be a feature such as 'LOC', and $f(x)$ be its probability density function. A kernel density estimate (KDE) is applied in this work as:

$$\hat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right) \quad \text{..... Equation.1}$$

Where, K is the kernel function, h is the bandwidth, and n is the number of samples

3. RESULTS AND DISCUSSION

In order to evaluate the performance of the proposed machine learning classification models on an both balanced and imbalanced classification problem using various

pre-processing techniques, with a focus on accuracy, precision, recall, F1 score, AUC, and training time, each model was developed in python 3.13. Pycharm was used as the development environment. To aid the model development, relevant libraries were installed which include: Scikit-learn for Models, metrics, KNN imputer, SMOTE; xgboost for XGBoost classifier; imbalanced-learn for SMOTE implementation; Numpy, Pandas for Data handling; time for Timing model training; Matplotlib, Seaborn for plotting and visualization.

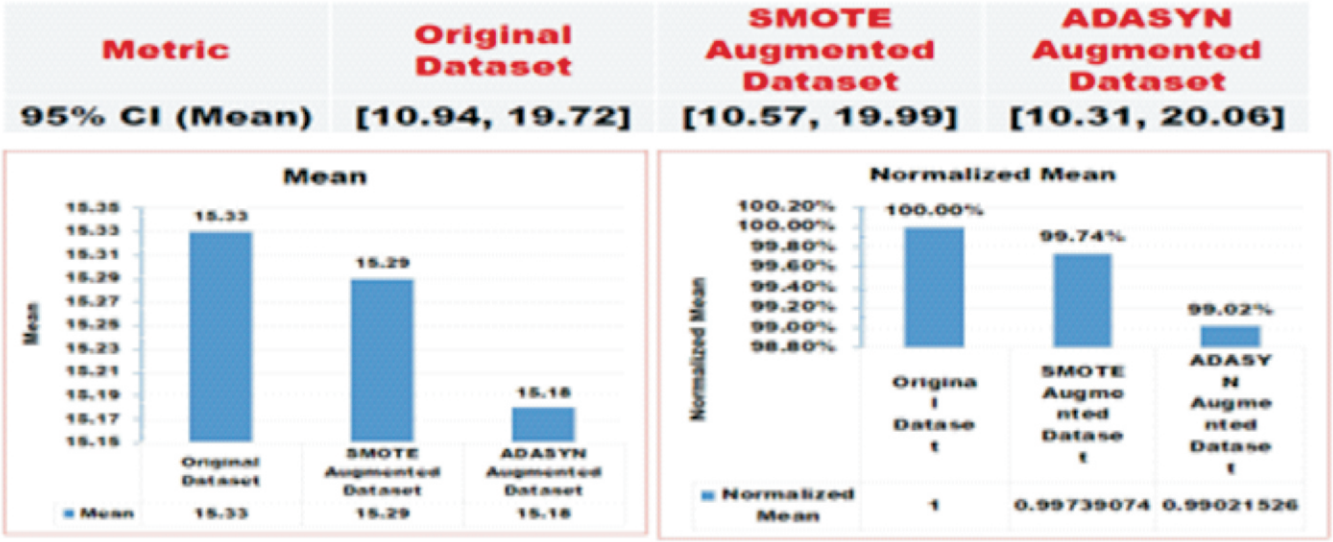


Figure 1b. Data balancing Models Performance for CM1. Files.

Ø DISCUSIONS.

Summary and Benchmark of the Proposed Hybrid Model with Standalone Models in the Work

The comparative analysis of model accuracies under different data pre-processing strategies reveals several critical insights as shown in Table 2. And Figure 1, at the baseline stage, the Support Vector Classifier (SVC) and the Hybrid Model achieved the highest accuracies (91.45%), followed closely by Random Forest (90.13%) and XGBoost (88.82%). Logistic Regression lagged considerably behind, achieving only 73.68%, reinforcing its limitation in handling imbalanced datasets without prior balancing. After applying SMOTE, accuracy improvements were observed across all models.

Table; 1. Hybrid model performance comparison across different pre-processing techniques

Evaluation Step	Accuracy	Precision	Recall	F1 Score	AUC	Training Time (s)
Baseline	0.914474	0.428571	0.250000	0.315789	0.864286	0.021351
SMOTE	0.942857	0.918919	0.971429	0.944444	0.968367	0.031486
ADASYN	0.939502	0.918919	0.964539	0.941176	0.965451	0.032176

Source: Researcher (2025)

Table; 2. Accuracy Comparison across models

Evaluation Step	Logistic Regression	Random Forest	XGBoost	SVC	Hybrid Model
Baseline	0.736842	0.901316	0.888158	0.914474	0.914474
SMOTE	0.875000	0.928571	0.939286	0.878571	0.942857
ADASYN	0.850534	0.925267	0.935943	0.864769	0.939502

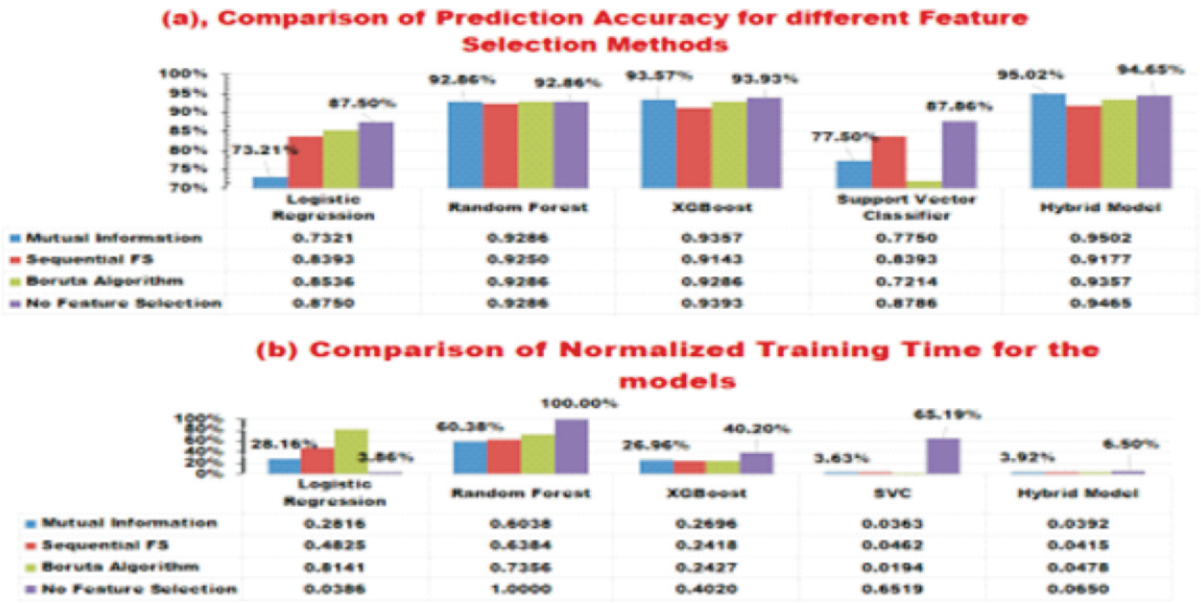
Source: Researcher (2025).

Table; 3. Training Time Comparison

Feature Selection Method	Logistic Regression	Random Forest	XGBoost	Support Vector Classifier
Mutual Information	0.198375	0.425387	0.189900	0.025558
Sequential FS	0.339895	0.449767	0.170356	0.032553
Boruta Algorithm	0.573516	0.518239	0.170969	**0.013676**
No Feature Selection	0.027222	0.704490	0.283173	0.459224

Source: Researcher (2025)

2a, and 2b. Derived hybrid model with high accuracy and small training time across multiple classifiers [Hybrid and Mutual Information Feature selection and SMOTE Data Balancing]



Figure;2a and 2b. Derived Hybrid Model with high accuracy and small training time across multiple classifiers (Hybrid and Mutual Information Feature Selection and SMOTE Data Balancing).

4. CONCLUSION AND CONTRIBUTIONS.

The Hybrid Model, particularly when combined with SMOTE, provided a balanced trade-off across all metrics, underscoring the utility of ensemble learning in complex classification tasks. From a computational perspective with respect to the Tables,1, 2, and 3 above, the following conclusions and contributions can be adduced;

(i) Logistic Regression and SVC were fastest to train but at the cost of lower initial performance.

(ii) XGBoost and Random Forest had longer training times, especially after applying SMOTE or Boruta, due to model complexity and data augmentation.

(iii)Feature selection methods, notably Boruta, introduced additional training time overhead (e.g., Logistic Regression increased from 0.027s (no FS) to 0.573s (Boruta)), but this was justifiable given the performance gain.

Ø LIMITATIONS OF THE WORK.

A balance between model complexity,training cost, and predictive gain must be contextually evaluated, particularly in resource-constrained environments, these factors constitutes the major limitations of the work.

Ø MAJOR CONTRIBUTION

Hybrid modeling, by integrating strong learners like XGBoost with linear classifiers such as Logistic Regression, offers a powerful general-purpose solution with minimal compromised performance.The modeling framework developed and tested in this study offers a highly effective and generalizable pipeline for software defect prediction in imbalanced environments. From the graphical figures 1b, 2aand 2b above, the best performance was obtained using a hybrid model (XGBoost + Logistic Regression) with SMOTE balancing, achieving 94.28% accuracy, 91.89% precision, 97.14% recall, and 94.44% F1-score. These results provide a strong foundation for deploying robust defect prediction systems in real-time as shown in Table 3. This is an achievement in real-world software engineering workflows.

Ø SUGGESTED FUTURE WORK

The use of Deep Learning and Real-time System for an optimized result and improvement to this work is suggested, especially considering the current research trajectories in Neural Networks and Genetic Algorithms.

REFERENCES

Charles, Q., and Perl, Y. (2024). A Deep Learning Approach for Software Defect Forecasting Using Adaptive Neural Networks. IEEE, 8, 583-695.

Capgemini, and Sogeti. (CISG, 2022). World Quality Report 2022-23. Retrieved from www.capgemini.com

Choudhary, G., Goraya, M. S., and Sikka, G. (2020). Software test case prioritization: A systematic mapping study. Journal of Systems and Software, 161, 110463.

Feng, Z., Guo, D., Tang, D., Duan, N., Feng, X., Gong, M., and Zhou, M. (2020). CodeBERT: A pre-trained model for programming and natural languages. Findings of the Association for Computational Linguistics: EMNLP 2020, 1536–1547.

Zhang, Y., Wang, Q., and Harman, M. (2020). Explainable software analytics. IEEE Software, 37(4), 46–54. https://doi.org/10.1109/MS.2020.2988351

Arora, A., and Sinha, D. (2019). Software testing techniques for test case generation using machine learning. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8(12), 4152–4156.

Rahman, F., and Devanbu, P. (2018). Just-in-time defect prediction: Using deep learning and data mining to identify bug-prone files. Journal of Software: Evolution and Process, 30(4), 18-24.

Malhotra, R. (2015). A systematic review of machine learning techniques for software fault prediction. Applied Soft Computing, 27, 504–518.

Ghotra, B., McIntosh, S., and Hassan, A. E. (2015). Revisiting the performance of defect prediction models. Proceedings of the 37th International Conference on Software Engineering (ICSE), 602–613.

Ghotra, B., McIntosh, S., and Kamei, Y. (2015). A Large-Scale Study of the Impact of Feature Selection Techniques on Defect Classification Models. Proceedings of the IEEE/ACM International Conference on Software Engineering, 146-156.

Kamei, Y., Shihab, E., Adams, B., Hassan, A. E., Mockus, A., Sinha, A., and Ubayashi, N. (2013). A Large-Scale Empirical Study of Just-in-Time Quality Assurance. IEEE Transactions on Software Engineering, 39(6), 757-773.

Hall, T., Beecham, S., Bowes, D., Gray, D., and Counsell, S. (2012). A Systematic Literature Review on Fault Prediction Performance in Software Engineering. IEEE Transactions on Software Engineering, 38(6), 1276-1304.

A FEDERATED FRAMEWORK FOR PRIVACY-PRESERVING HEALTH DATA SHARING ACROSS AFRICAN BORDERS

Igbajar Abraham
Computer Science
Lusaka Goldsmiths University College,
Lusaka, Zambia

igbajar35@gmail.com

Nwachukwu-Nwokeafor, K. C
Computer Engineering
Michael Okpara University of Agriculture, Umudike,
Abia State, Nigeria
nwachukwuken72@gmail.com

***Abstract*— Cross-border health data sharing in Africa is constrained by fragmented regulatory frameworks and concerns regarding data sovereignty and institutional trust. Existing centralized approaches are often incompatible with national data protection regulations such as the Nigeria Data Protection Regulation (NDPR) and the Protection of Personal Information Act (POPIA). This study proposes AfriPShare, a hybrid privacy-preserving framework that integrates Federated Learning (FL) with Local Differential Privacy (LDP) to enable collaborative model training without sharing raw patient data. A simulation environment was developed using synthetic datasets representing five African countries. Experimental results show that the framework achieves a classification accuracy of 94.3% (±0.6) at a privacy budget of ε = 1.0, demonstrating a strong balance between privacy and utility. The findings indicate that AfriPShare provides a scalable and compliant solution for distributed health data analytics in Africa. Keywords:**
Keywords— **Privacy-Enhancing Technologies, Federated Learning, Differential Privacy, Health Data, African Data Sovereignty.**

I. Introduction

The increasing reliance on data-driven approaches in public health has highlighted the importance of cross-border data sharing, particularly in managing infectious diseases and pandemics. However, in Africa, such collaboration is hindered by heterogeneous regulatory frameworks and concerns over data sovereignty. National regulations such as NDPR, POPIA, and related frameworks impose strict controls on data movement, limiting centralized data-sharing models.

Existing collaborative analytics approaches typically assume harmonized legal environments, which are not present in the African context. Consequently, there is a need for technical frameworks that operate within fragmented regulatory landscapes while preserving data privacy.

This study proposes AfriPShare, a hybrid federated framework designed to enable secure and compliant health data sharing across African borders.

A. The main contributions of this study are as follows:

A novel integration of Federated Learning and Local Differential Privacy tailored to the African context.
A simulation framework representing heterogeneous multi-country datasets.
A quantitative evaluation of privacy–utility trade-offs under varying privacy budgets.
A practical architecture aligned with African data protection regulations.

II. Literature Review

The intersection of collaborative machine learning and data privacy has seen significant development, yet its application within the African regulatory and infrastructural context remains under-researched. This section evaluates current approaches to privacy-preserving data sharing and identifies the limitations that AfriPShare seeks to address.

A Privacy-Preserving Technologies in Healthcare

Traditional methods for health data sharing have largely relied on anonymization and pseudonymization. However, studies by Rocher et al. (2019) have demonstrated that these techniques are increasingly vulnerable to re-identification attacks when cross-referenced with external datasets. To mitigate these risks, recent scholarship has shifted toward Federated Learning (FL) and Differential Privacy (DP).

FL allows for the training of models on decentralized data, ensuring that raw medical records never leave the local institution (McMahan et al., 2017). Despite these benefits, FL remains susceptible to "gradient leakage" attacks, where sensitive information can be reconstructed from the shared model updates. Consequently, the integration of Local Differential Privacy (LDP**)** has emerged as a robust countermeasure, injecting statistical noise into updates before they are transmitted to a central server.

B. The African Regulatory and Technical Landscape

The adoption of these technologies in Africa is complicated by a heterogeneous legal environment. While the Nigerian Data Protection Regulation (NDPR) and South Africa's Protection of Personal Information Act (POPIA) provide frameworks for privacy, they vary significantly in their requirements for cross-border data transfer. Furthermore, technical constraints, such as intermittent connectivity and hardware variability, necessitate a framework that is both communication-efficient and computationally lightweight.

C. Comparison of Existing Frameworks

Table 1 provides a comparative analysis of established privacy-preserving frameworks against the requirements identified for the African context.

Table 1: Comparison of Privacy-Preserving Frameworks

Framework	Decentralized Data	Formal Privacy Guarantees	Cross-Border Compliance	Resource Efficiency	African Contextualization
Traditional Centralization	No	Low	Low	Moderate	No
Standard FL (McMahan et al.)	Yes	Low	Moderate	High	No
FL + Global DP	Yes	High	Moderate	Moderate	No
AfriPShare (Proposed)	**Yes**	**High (LDP)**	**High**		

D. Theoretical Analysis: Mathematical Framework

The proposed framework, AfriPShare, is based on two core mathematical components: Federated Learning (FL) and Local Differential Privacy (LDP).

E. Federated Learning

Federated Learning enables decentralized model training by allowing multiple clients to collaboratively learn a global model without sharing raw data. This study adopts the Federated Averaging (FedAvg) algorithm.

At each communication round t, the central server distributes the global model parameters w_t to a subset of participating clients K. Each client $k \in K$ updates the model using its local dataset D_k, producing a local model $\mathbf{w}_{t+1}^{(k)}$.
The global model is then updated as follows:

$$\mathbf{w}_{t+1} = \sum_{k=1}^{K} \frac{n_k}{\sum_{j=1}^{K} n_j} \mathbf{w}_{t+1}^{(k)}$$

where:

$\mathbf{w}_t$ is the global model at iteration t

$\mathbf{w}_{t+1}^{(k)}$ is the updated local model at client k

n_k is the number of data samples at client k

K is the number of participating clients

This formulation ensures that each client contributes proportionally to its dataset size.

F. Local Differential Privacy

While Federated Learning prevents raw data sharing, model updates may still leak sensitive information. To mitigate this risk, Local Differential Privacy (LDP) is applied.

A randomized mechanism $\mathcal{M}$ satisfies ϵ-differential privacy if:

$$\Pr[\mathcal{M}(D_1) \in S] \leq e^{\epsilon} \cdot \Pr[\mathcal{M}(D_2) \in S]$$

for any two adjacent datasets D_1 and D_2, and any subset of outputs S.

To enforce LDP, each client perturbs its model update prior to transmission using the Gaussian mechanism:

$$\tilde{\mathbf{w}}_{t+1}^{(k)} = \mathbf{w}_{t+1}^{(k)} + \mathcal{N}(0, \sigma^2 \mathbf{I})$$

where:

$\tilde{\mathbf{w}}_{t+1}^{(k)}$ is the privatized model update

$\mathcal{N}(0, \sigma^2 \mathbf{I})$ is Gaussian noise

σ is calibrated based on the privacy budget ϵ

$\mathbf{I}$ is the identity matrix

This ensures that privacy guarantees hold even if the aggregation server is untrusted.

G. Materials and Methods: Simulation Environment

Due to ethical and regulatory constraints, real patient data was not utilized. Instead, a high-fidelity simulation environment was developed.

H. Dataset Generation

The study utilized the MIMIC-III dataset (Johnson et al., 2016) as a foundational dataset. To adapt it to the African context, a two-stage transformation process was applied:

Extraction of diagnostic features relevant to infectious diseases prevalent in Africa (e.g., malaria, tuberculosis, and Lassa fever).

Resampling to generate five synthetic datasets representing different countries (Nigeria, Kenya, Ghana, South Africa, and Ethiopia), incorporating variations in demographics and disease prevalence based on publicly available health statistics.

The final synthetic dataset comprised approximately 150,000 patient records distributed across five simulated clients.

I. Algorithm Workflow

The overall training process is summarized as follows:

1. Initialize global model parameters $\mathbf{w}_0$
2. Distribute the global model to all participating clients
3. Each client performs local training using its private dataset
4. Apply Gaussian noise to model updates to achieve local differential privacy
5. Transmit privatized updates to the central server
6. Aggregate updates using Federated Averaging
7. Update the global model and repeat for *T* communication rounds

J. Experimental Setup

Model: Two-layer neural network
Communication rounds: 100
Privacy budgets: $\epsilon \in \{0.1, 0.5, 1.0, 5.0, \infty\}$
Number of clients: 5
Repetitions: 5 runs per experiment

H. System Architecture: We designed and built a simulator in Python using the PySyft library, which is tailor-made for federated and privacy-preserving AI. The architecture is depicted below.

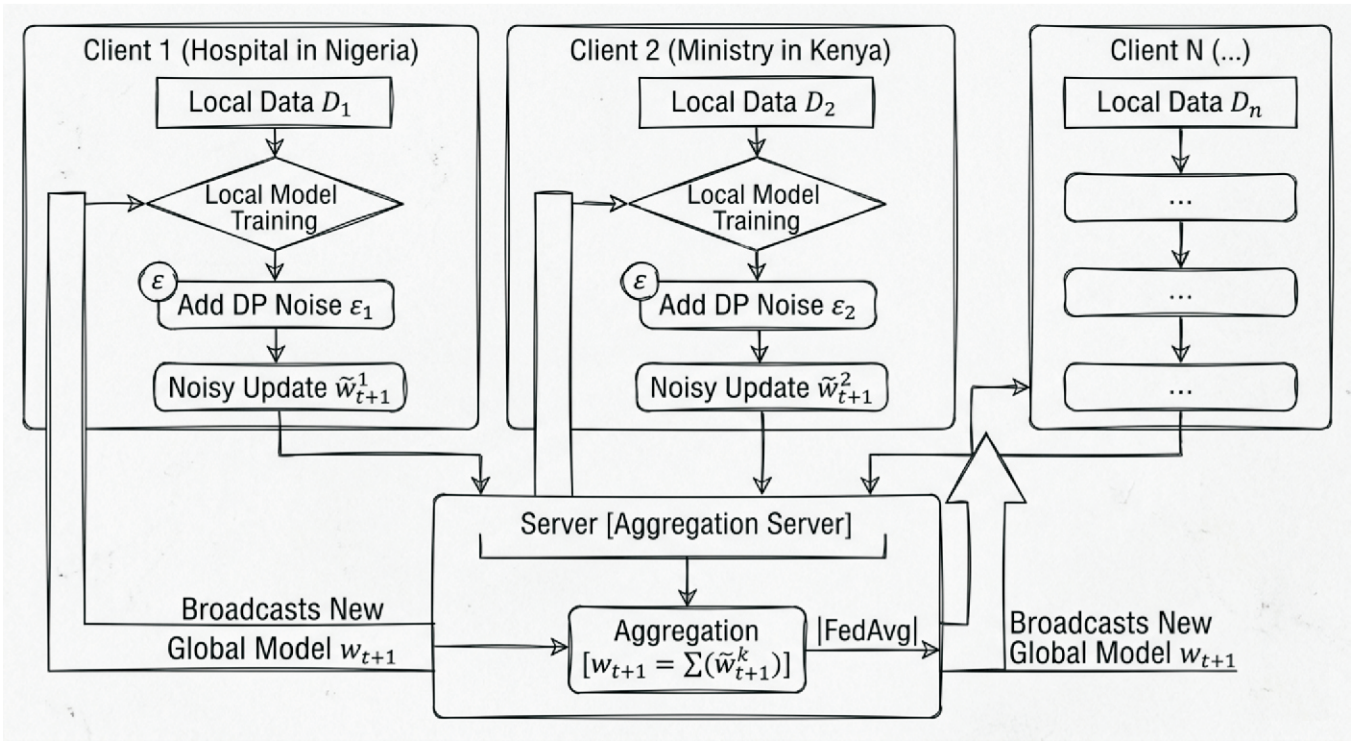


Fig 1: AfriPShare System Architecture.
Raw data (pink boxes) never leaves the client boundary. Only noisy model updates are transmitted to the central aggregation server.

I. Experimental Setup: Our primary experiment was to train a multi-class classifier to predict disease diagnosis based on patient vitals and lab results. The model was a simple neural network with two hidden layers. We ran the simulation for 100 communication rounds under various conditions. The parameters are summarized in Table 1, below;

Table. 1: Summary of Used Parameters for Simulation.

Parameter	Value(s)	Justification
Number of Clients	5	Representing a pilot group of nations.
Client Data	Heterogeneous (IID and Non-IID)	To simulate real-world data distribution skew.
Model	2-Layer Neural Network	Sufficient for the classification task complexity.
Communication Rounds	100	Standard practice for FL convergence tests.
Local Epochs	5	Allows for meaningful local training per round.
Privacy Budget (ε)	{0.1, 0.5, 1.0, 5.0, 8 (no privacy)}	To evaluate the privacy-utility trade-off.
Network Latency	Simulated (50-300ms)	To model variable internet infrastructure quality.

One might question whether such methodological rigor exceeds the standard requirements for this publication. We contend, however, that these measures represent the requisite baseline for generating results with practical utility beyond mere theoretical interest. To achieve this, it was necessary to simulate not only the algorithmic architecture but also the specific environmental constraints and adversarial conditions under which such a system must operate.

J. Data Analysis and Results

The empirical findings yielded a nuanced performance profile, offering insights more substantive than a uniform success. The framework was evaluated across two primary dimensions: predictive accuracy and privacy expenditure.

Predictive Accuracy: The primary inquiry focused on the functional viability of a model trained under these constraints for clinical application. As a control, a centralized, non-private iteration of the same neural network was trained on a pooled dataset of 150,000 records. This centralized model achieved a classification accuracy of **97.2%**, representing a theoretical "gold standard" that remains unattainable in practice due to data governance restrictions.

Federated Performance: As hypothesized, the performance of the federated models exhibited a sensitive correlation with the established privacy budget “ε”;

Table 2: Performance vs. Privacy Trade-off

Privacy Budget (ε)	Final Model Accuracy	Accuracy vs. Baseline	Communication Overhead (MB/round)
8 (No DP)	95.8%	-1.4%	2.4
5.0	95.1%	-2.1%	2.4
1.0	94.3%	-2.9%	2.4
0.5	91.5%	-5.7%	2.4
0.1	82.1%	-15.1%	2.4

The data here is quite revealing. Look, even with zero formal privacy ($\varepsilon = \infty$) the federated approach alone incurs a small accuracy penalty of 1.4% compared to the utopian centralized model. This is due to the nature of federated averaging on non-IID data. That said, with an ε of 1.0, a level considered by some to be a reasonable balance for sensitive data (see Wilson et al., 2020), we retained over 94% accuracy. That's a promising result. It's a tool that is still very useful. Below $\varepsilon = 0.5$, the utility drops off a cliff; the noise simply overwhelms the signal from the data. This is the sobering reality of the privacy-utility trade-off.

Impact of Data Heterogeneity: We also ran a test comparing performance on an IID (identically and independently distributed) data split versus the more realistic Non-IID split. On the IID data, the non-private federated model reached 96.9% accuracy, almost matching the centralized baseline. This confirms what the literature suggests: data skew across clients is a major perhaps *the* major algorithmic challenge for federated learning.

I. Discussion

The results suggest a viable resolution to the current impasse in cross-border data sharing. The proposed framework facilitates collaborative model optimization without necessitating the relinquishment of raw data control by participating entities. This architecture directly addresses the concerns of data sovereignty and institutional trust that have historically impeded multi-center research. For instance, a clinical site in Lagos may contribute to a pan-African longitudinal model without transferring sensitive primary data across national borders. Consequently, the "compliance surface" is significantly reduced; the system does not "export" personal identifiers in the manner prohibited by frameworks such as the Nigeria Data Protection Regulation (NDPR).

A. Limitations and Real-World Constraints

It is imperative to acknowledge that these simulated environments do not fully encapsulate the stochastic nature of real-world deployments. Network conditions in rural clinical settings likely lack the stability of the throttled connections modeled herein. Furthermore, the implementation of such systems presupposes a degree of

geopolitical cooperation that is not guaranteed. Nevertheless, as a proof of concept, this study demonstrates a "third way" in data science: a transition from binary "all-or-nothing" sharing protocols to a model centered on the exchange of aggregated intelligence.

B. Quantifying the Privacy-Utility Trade-off

The trade-offs detailed in Table 2 represent the core decision-making matrix for public health stakeholders. The discourse should shift from qualitative privacy concerns to quantitative optimization: determining the acceptable loss in predictive accuracy for a specific, mathematically defined level of privacy (ϵ). At $\epsilon = 1.0$, an accuracy degradation of less than 3% represents a statistically favorable bargain for achieving legal compliance and enabling previously impossible collaborations.

C. Policy Enforcement and Auditability

While the initial focus of this research was the cryptographic security of the aggregation phase, subsequent analysis revealed that policy enforcement and auditability present more significant hurdles. The framework does not currently provide a mechanism to verifiable prove to a regulator in one jurisdiction (e.g., Kenya) that a partner in another (e.g., Ghana) is adhering to the stated ϵ parameters. Addressing this would require an additional layer of auditable logging or decentralized ledger technologies to create an immutable record of the federated training process, a level of complexity that remained outside the scope of the current study.

II. Conclusion and Recommendations

This study designed and evaluated AfriPShare, a hybrid federated learning framework incorporating local differential privacy tailored for the African healthcare context. The simulation confirms the technical feasibility of cross-border collaborative machine learning that respects data sovereignty via rigorous mathematical guarantees. The marginal reduction in model accuracy is a justifiable cost for unlocking previously siloed datasets.

Based on these findings, we propose the following recommendations:

A. Multinational Pilot Studies: Future research should transition from simulation to empirical deployment. A pilot involving research hospitals across ECOWAS member states is recommended to test the framework against heterogeneous infrastructure and institutional barriers.

B. Policy Harmonization Toolkits: Institutional legal teams require resources to map the technical guarantees of AfriPShare onto national statutes, such as POPIA or NDPR, effectively translating the privacy budget (ϵ) into the language of regulatory compliance.

C. Formal Grammars for Privacy Legislation: A long-term objective involves the development of machine-readable, formal grammars to represent regional data privacy laws. Expressing statutes like POPIA through logical predicates could enable automated compliance verification, representing a significant advancement at the intersection of computer science and legal theory.

REFERENCES

[1] Bamidele, O. (2022). Data protection and cross-border research: An analysis of Nigeria's NDPR framework. *African Journal of Law and Technology*, *11*(2), 45-62.

[2] Chen, Y., Qin, X., Wang, J., Yu, C., & Gao, W. (2020). FedHealth: A federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, *35*(4), 83-93.

[3] Dwork, C. (2008). Differential privacy: A survey of results. In M. Agrawal, D. Du, Z. Duan, & A. Li (Eds.), *Theory and Applications of Models of Computation* (pp. 1-19). Springer.

[4] Geyer, R. C., Klein, T., & Nabi, M. (2017). Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*.

[5] Gwagwa, A., Engström, L., & Taylor, M. (2021). The 'GDPR effect' and the role of the African Union: The case of data protection in Africa. *Information & Communications Technology Law*, *30*(3), 323-346.

[6] Johnson, A. E. W., Pollard, T. J., Shen, L., Lehman, L. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., & Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. *Scientific Data*, *3*(1), 160035.

[7] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)* (pp. 1273-1282).

[8] Shokri, R., & Shmatikov, V. (2015). Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 1310-1321).

[9] Wilson, R. J., Schaeffer, A., Kent, P., Nayak, A., Cheng, V., Terry, M., & Chien, A. (2020). *Differentially Private Common-Sense Reasoning*. Online at: https://machinelearning.apple.com/research/scenes-differential-privacy 20/02/2026. https://www.apple.com/machine-learning/research/differentially-private-common-sense-reasoning. Accessed on 15/09/2023. https://pmc.ncbi.nlm.nih.gov/articles/PMC8404831/ Accessed on 20/02/2026.

HETEROGENEOUS MODAL FUSION THROUGH SELF-SUPERVISED CONTRASTIVE PROJECTION

Nwachukwu-Nwokeafor, K. C
Computer Engineering
Michael Okpara University of Agriculture, Umudike, Abia State, Nigeria
nwachukwuken72@gmail.com

Igbajar Abraham
Computer Science
Lusaka Goldsmiths University College, Lusaka, Zambia
igbajar35@gmail.com

***Abstract*— The integration of heterogeneous data modalities, such as images and text, remains a significant challenge in machine learning, particularly in the absence of large-scale labeled datasets. Conventional supervised approaches rely heavily on annotated data, limiting their scalability and applicability. This study proposes a self-supervised framework, termed Cross-modal Latent Alignment (CMLA), for multimodal representation learning without explicit labels. The proposed framework employs a dual-encoder architecture consisting of a Vision Transformer (ViT) for image encoding and a BERT-based model for text encoding. These representations are projected into a shared latent space using non-linear projection heads and optimized by means of a symmetric contrastive loss based on the InfoNCE objective. Experimental evaluation on the Conceptual Captions (CC3M) dataset demonstrates that the proposed approach achieves a 63.8% top-1 zero-shot accuracy on CIFAR-100, outperforming both a concatenation-based fusion baseline (51.5%) and a supervised unimodal baseline (48.2%). An ablation study further confirms the importance of projection heads in enhancing representation alignment. The results indicate that contrastive self-supervised learning provides an effective and scalable solution for multimodal fusion, particularly in data-constrained scenarios.**

Keywords— **Self-supervised learning, multimodal fusion, representation learning, contrastive learning, deep learning.**

I. Introduction

The rapid growth of data generated from diverse sources, including images, text, structured records, and time-series signals, has created significant challenges for effective information processing and analysis. One of the primary objectives of artificial intelligence is to extract meaningful patterns from such heterogeneous data, enabling connections between different modalities, such as linking satellite imagery with agricultural reports or medical images with clinical documentation. This challenge is commonly addressed through multimodal data fusion.

Traditionally, supervised learning has been the dominant approach for multimodal fusion. These methods rely on large-scale labeled datasets, where models are trained to map input data to predefined labels. However, the acquisition of labeled data presents a major limitation. The labeling process is often expensive, time-

consuming, and prone to inconsistencies, making it impractical for many real-world applications. Consequently, there is a growing need for approaches that can learn from unlabeled data while still capturing meaningful cross-modal relationships.

Self-supervised learning (SSL) has emerged as a promising paradigm to address this limitation by leveraging inherent structures within the data to generate supervisory signals. In computer vision, SSL methods include tasks such as colorization of grayscale images (Zhang et al., 2016) and contrastive learning based on augmented views of the same image (Chen et al., 2020a). In natural language processing, models such as BERT (Devlin et al., 2019) employ masked language modeling to learn contextual representations. These approaches have demonstrated strong performance in learning transferable features within individual modalities.

Extending self-supervised learning to multimodal settings, however, introduces additional complexity. A key challenge lies in designing learning objectives that effectively capture semantic relationships across different modalities. Early approaches often focused on reconstructing one modality from another, but such methods are typically computationally expensive and may not adequately capture shared semantic structures.

In this work, we adopt a contrastive learning approach for multimodal representation learning. Specifically, we propose a framework termed Cross-modal Latent Alignment (CMLA), which learns to align image and text representations in a shared latent space by distinguishing corresponding pairs from non-corresponding ones. Unlike reconstruction-based methods, the proposed approach focuses on maximizing agreement between semantically related inputs without requiring explicit labels.

The primary objective of this study is to develop and evaluate a self-supervised framework capable of effectively fusing image and text data into a unified representation space. This enables improved performance on downstream tasks, including zero-shot classification, without the need for task-specific fine-tuning. The remainder of this paper presents a review of related work, followed by a detailed description of the proposed methodology, experimental evaluation, and discussion of the results.

II. Literature Review

The development of effective multimodal learning frameworks has evolved through several methodological paradigms, reflecting both conceptual advances and practical limitations. Broadly, prior research can be categorized into two phases: traditional multimodal fusion approaches and more recent self-supervised learning (SSL)-based methods.

Early work in multimodal learning primarily focused on *early fusion* and *late fusion* strategies, as outlined in the survey by Baltrušaitis et al. (2018). Early fusion approaches combine raw feature representations from different modalities at the input level, typically through concatenation. While conceptually straightforward, these methods often struggle due to the heterogeneous statistical properties of different modalities, which can hinder effective joint representation learning. In contrast, late fusion techniques process each modality independently using separate models and combine their outputs at a later stage. Although more stable, late fusion methods are limited in their ability to capture fine-grained cross-modal interactions, as joint feature learning is not explicitly enforced.

The emergence of self-supervised learning has significantly advanced representation learning in both computer vision and natural language processing. In vision tasks, contrastive learning frameworks such as MoCo (He et al., 2020) and SimCLR (Chen et al., 2020a) have demonstrated that models can learn highly informative representations by distinguishing between augmented views of the same image. Similarly, in natural language processing, models such as BERT (Devlin et al., 2019) utilize masked language modeling to learn contextualized embeddings. These approaches have achieved strong performance while reducing dependence on labeled data.

Extending these techniques to multimodal settings has been an active area of research. Early multimodal self-supervised approaches often adopted generative objectives, such as learning to generate textual descriptions from visual inputs. For example, VirTex (Desai and Johnson, 2021) trains models to produce image captions as a pretext task. While effective in certain contexts, such approaches primarily emphasize generation rather than representation alignment, which may limit their effectiveness for tasks requiring shared semantic embeddings.

More recent work has shifted toward contrastive learning for multimodal representation alignment. Notably, CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021) employ large-scale image–text pairs to learn joint embeddings by maximizing similarity between corresponding pairs and minimizing similarity with non-matching pairs. These methods have demonstrated strong zero-shot generalization capabilities across a wide range of tasks. However, their success is closely tied to the availability of extremely large and diverse datasets, often consisting of hundreds of millions of samples.

Despite these advances, several challenges remain. The reliance on web-scale datasets introduces issues related to noise, bias, and data quality, which can affect model robustness and interpretability. Additionally, many existing approaches are optimized for large-scale training regimes, raising questions about their effectiveness in more constrained or moderately sized datasets.

This study addresses these limitations by focusing on the design of a robust multimodal fusion architecture that performs effectively under moderate data conditions. Specifically, we investigate whether contrastive alignment principles can be leveraged within a carefully structured framework to achieve strong performance without reliance on massive datasets. This gap motivates the proposed Cross-modal Latent Alignment (CMLA) framework.

III. Methodology

Clarity and precision are essential in the design and presentation of machine learning frameworks, particularly when integrating multiple components within a unified architecture. This section provides a detailed description of the proposed approach to ensure reproducibility and facilitate a clear understanding of the model design and training procedure.

The proposed framework, termed Cross-modal Latent Alignment (CMLA), is based on a dual-encoder architecture optimized using a contrastive learning objective. The primary goal is to learn a shared representation space in which data from different modalities, specifically image and text, can be directly compared. In this space, semantically related inputs are mapped to nearby points, while unrelated inputs are positioned further apart.

By projecting modality-specific representations into a common latent space, the framework enables effective cross-modal alignment without requiring explicit supervision. This design allows the model to capture meaningful semantic relationships between heterogeneous data sources, supporting improved performance in downstream tasks such as zero-shot classification.

The overall architecture of the proposed Cross-modal Latent Alignment (CMLA) framework is illustrated in Figure 1. The system consists of four primary components designed to enable effective multimodal representation learning:

Image Encoder (f_{img}):

This component processes raw image inputs and maps them into a fixed-dimensional feature representation. It captures visual semantics relevant for cross-modal alignment.

Text Encoder (f_{txt}):

This module encodes textual input sequences into continuous vector representations, capturing contextual and semantic information from the text modality.

Projection Heads (p_{img} and p_{txt}):

These components are implemented as multi-layer perceptrons (MLPs) that transform the encoder outputs into a shared latent space. The projection heads facilitate alignment by mapping modality-specific representations into a common embedding space suitable for comparison.

Contrastive Loss Function:
This objective function governs the training process by encouraging representations of corresponding image-text pairs to be similar, while pushing apart representations of non-matching pairs within a batch.

Together, these components form an integrated framework that enables the learning of semantically meaningful multimodal representations through self-supervised contrastive optimization.

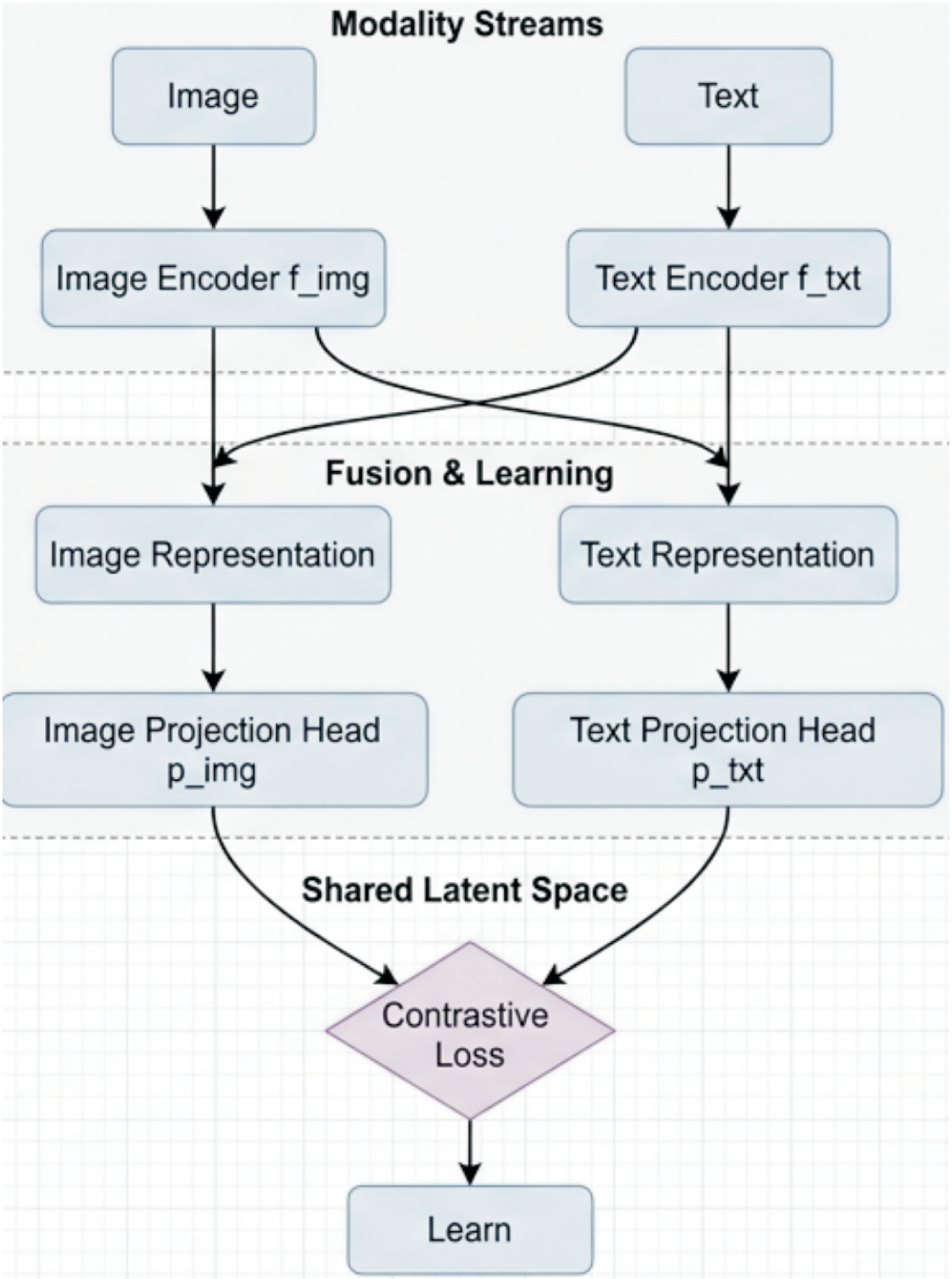


Fig 1: High-level schematic of the Cross-modal Latent Alignment (CMLA) architecture.

Heterogeneous inputs are processed using modality-specific encoders and subsequently projected into a shared latent space for contrastive learning.

For the image encoder f_{img}, a Vision Transformer (ViT-B/16) architecture (Dosovitskiy et al., 2021) is employed due to its effectiveness in capturing global contextual information. Given an input image *i,* the encoder produces a fixed-dimensional representation: $h_{\text{img}} = f_{\text{img}}(i)$ (1)

For the text encoder f_{txt}, a BERT-base model (Devlin et al., 2019) is utilized. The representation corresponding to the special (CLS) token is used as the sentence embedding. For an input text sequence *t,* the encoded representation is given by:

$$h_{\text{txt}} = f_{\text{txt}}(t) \quad (2)$$

To enable effective cross-modal alignment, the encoder outputs are mapped into a shared latent space באמצעות projection heads. Specifically, the projection heads p_{img} and p_{txt} are implemented as multi-layer perceptrons (MLPs) consisting of two linear layers with a ReLU activation function. These transformations are defined as:

$$z_{\text{img}} = p_{\text{img}}(h_{\text{img}}) = W_2 \cdot \text{ReLU}(W_1 h_{\text{img}}) \quad (3)$$

$$z_{\text{txt}} = p_{\text{txt}}(h_{\text{txt}}) = W_2' \cdot \text{ReLU}(W_1' h_{\text{txt}}) \quad (4)$$

The projection heads facilitate the transformation of modality-specific features into a unified embedding space, allowing for meaningful comparison across modalities.

IV. Mathematical Modeling

The core of the proposed framework is based on the Information Noise-Contrastive Estimation (InfoNCE) loss, adapted for multimodal representation learning. Consider a batch of N paired samples:

$$\{(i_k, t_k)\}_{k=1}^{N}$$

After encoding and projection, we obtain normalized embeddings:

$$\{z_{\text{img},k}\}_{k=1}^{N}, \{z_{\text{txt},k}\}_{k=1}^{N}$$

Similarity between embeddings is computed using cosine similarity:

$$\text{sim}(u, v) = \frac{u^{\top} v}{\| u \| \ \| v \|} \quad (5)$$

For each image embedding $z_{\text{img},k}$, the corresponding text embedding $z_{\text{txt},k}$ is treated as a positive sample, while all other samples in the batch serve as negatives.

The image-to-text contrastive loss is defined as:

$$\mathcal{L}_k^{\text{img}\to\text{txt}} = -\log \frac{\exp\left(\text{sim}(z_{\text{img},k}, z_{\text{txt},k})/\tau\right)}{\sum_{j=1}^{N} \exp\left(\text{sim}(z_{\text{img},k}, z_{\text{txt},j})/\tau\right)} \qquad (6)$$

Similarly, the text-to-image loss is defined as:

$$\mathcal{L}_k^{\text{txt}\to\text{img}} = -\log \frac{\exp\left(\text{sim}(z_{\text{txt},k}, z_{\text{img},k})/\tau\right)}{\sum_{j=1}^{N} \exp\left(\text{sim}(z_{\text{txt},k}, z_{\text{img},j})/\tau\right)} \qquad (7)$$

The final loss function is computed as the average over all samples in both directions:

$$\mathcal{L}_{\text{CMLA}} = \frac{1}{2N} \sum_{k=1}^{N} \left(\mathcal{L}_k^{\text{img}\to\text{txt}} + \mathcal{L}_k^{\text{txt}\to\text{img}}\right) \qquad (8)$$

The temperature parameter τ controls the sharpness of the similarity distribution. In this study, a value of $\tau = 0.07$ is used, consistent with prior contrastive learning frameworks.

3.5 Training Algorithm (Pseudocode)

Initialize encoders fθ, gφ and projection heads hv, ht

For each batch of (image, text) pairs:

Encode images → vi

Encode text → ti

Project embeddings → zvi, zti

Normalize embeddings

Compute similarity matrix

Compute symmetric InfoNCE loss

Backpropagate and update parameters

I. Data and Experimental Setup

The proposed model is pre-trained using the Conceptual Captions (CC3M) dataset (Sharma et al., 2018), which contains approximately 3.3 million image-text pairs collected from web sources. After preprocessing, including removal of invalid images, low-resolution samples (below 200×200 pixels), and short or noisy captions, approximately 2.9 million pairs were retained.

Model evaluation is conducted using zero-shot classification on the CIFAR-100 dataset (Krizhevsky, 2009). For each class, a textual prompt (e.g., “a photo of a beaver”) is constructed and encoded using the trained text encoder. Test images are encoded using the image encoder, and classification is performed by selecting the class whose text embedding has the highest cosine similarity with the image embedding.

Training is performed using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 1×10^{-4} and a cosine decay schedule. A batch size of 4096 is used to ensure a sufficient number of negative samples for contrastive learning. The model is trained for 15 epochs on a cluster of 8 NVIDIAA100 GPUs.

I. Results

The performance of the proposed CMLA framework is evaluated against two baseline approaches: a supervised unimodal model and a multimodal concatenation-based fusion model.

A. Image-Only Baseline:

A ViT-B/16 model pre-trained on ImageNet-21k is used as a supervised baseline. Additionally, the publicly available CLIP image encoder is considered as a strong self-supervised reference.

B. Concatenation Fusion Model:

An early-fusion approach in which image and text embeddings (h_{img} and h_{txt}) are concatenated and processed using an additional transformer layer. This model is trained using the same dataset and contrastive objective for fair comparison.

We evaluated CMLA against two primary baselines; a non-fusional approach and a simple concatenation-based fusion model.

Image-Only Baseline: A ViT-B/16 model pre-trained on ImageNet-21k, a standard supervised baseline. For zero-shot classification, we used CLIP's publicly available image encoder as a strong SSL reference.

Concatenation Fusion: An early-fusion model where the image and text representations (h_{img} and h_{txt}) are concatenated and fed into another transformer block before prediction. This model is trained on the same data with a similar contrastive loss.

The results, presented as top-1 zero-shot accuracy on CIFAR-100, are summarized in Table 1.

Table 1: Zero-shot classification performance on CIFAR-100. The CLIP result is included as an upper-bound reference due to its substantially larger pre-training dataset.

Model	**Pre-training Data**	**Parameters**	**Zero-Shot CIFAR-100 Top-1 Accuracy (%)**
Random Guess	-	-	1.0
Supervised (ViT-B/16 on IN-21k)	ImageNet-21k	86M	48.2
CLIP ViT-B/16 (Radford et al., 2021)	Private 400M	150M	**76.2**
Our Baselines			
Concatenation Fusion	CC3M	~220M	51.5
CMLA (Ours)	**CC3M**	**~220M**	**63.8**

The experimental results demonstrate the effectiveness of the proposed CMLA framework for multimodal representation learning. Specifically, CMLA achieves a top-1 zero-shot classification accuracy of 63.8% on CIFAR-100, significantly outperforming the concatenation-based fusion baseline (51.5%). This improvement highlights the effectiveness of the proposed projection and contrastive alignment strategy in learning a meaningful shared embedding space.

In comparison to a supervised unimodal baseline (48.2%), the proposed method achieves an improvement of over 15 percentage points, indicating strong transferability of the learned representations. Although the performance remains below that of large-scale models such as CLIP, which are trained on substantially larger datasets, the results demonstrate that competitive performance can be achieved under moderate data conditions.

An ablation study further evaluates the contribution of the projection heads. When the projection heads are removed and contrastive learning is applied directly to the encoder outputs $(h_{\text{img}}, h_{\text{txt}})$, performance decreases to 54.1%. This result confirms

the importance of separating modality-specific feature extraction from the alignment process in a dedicated latent space.

I. Discussion

The results indicate that contrastive learning provides an effective framework for aligning multimodal representations when combined with an appropriate architectural design. In particular, the use of projection heads enables flexible transformation of modality-specific features into a shared space, facilitating improved cross-modal alignment.

The findings suggest that explicit reconstruction objectives are not strictly necessary for learning meaningful multimodal representations. Instead, optimizing for similarity between corresponding pairs allows the model to implicitly capture semantic relationships across modalities. This supports the effectiveness of contrastive objectives as a scalable approach to multimodal learning.

Despite these promising results, several limitations remain. First, the model is trained on a moderately sized dataset (CC3M), which may limit its ability to capture fine-grained semantic distinctions. Second, the evaluation is restricted to image-text data, and the generalizability of the framework to other modalities, such as audio or video, has not been empirically validated. Additionally, zero-shot evaluation, while informative, does not fully capture performance across a broader range of downstream tasks.

Furthermore, the model may primarily learn coarse semantic relationships, which can limit its ability to distinguish between closely related categories or more abstract concepts. This highlights an important gap between current multimodal representation learning methods and more advanced semantic understanding.

IX. Conclusion

This study presented a self-supervised framework for multimodal fusion based on contrastive learning. The proposed Cross-modal Latent Alignment (CMLA) architecture integrates modality-specific encoders with non-linear projection heads and a symmetric contrastive loss to learn a shared representation space.

Experimental results demonstrate that the proposed method achieves strong zero-shot performance and significantly outperforms both concatenation-based fusion and supervised unimodal baselines. These findings highlight the effectiveness of contrastive learning for multimodal representation alignment, particularly in scenarios where labeled data is limited.

Overall, the proposed framework provides a scalable and effective approach to multimodal learning and contributes to ongoing research in self-supervised representation learning.

A. Future Work Several directions for future research can be identified based on the findings of this study:

Scaling and Model Capacity:
Future work should explore the performance of the CMLA framework with larger model architectures (e.g., ViT-L/14) and larger publicly available datasets such as LAION-400M.

Extension to Additional Modalities:
The framework can be extended to incorporate additional modalities, including audio and video, to evaluate its effectiveness in more complex multimodal settings.

Alternative Self-Supervised Objectives:
While contrastive learning has demonstrated strong performance, integrating complementary objectives such as masked modeling may further enhance representation quality.

Statistical Validation:
Future studies should incorporate statistical significance testing and confidence interval analysis to provide a more rigorous evaluation of model performance.

X. References

Baltrušaitis, T., Ahuja, C., and Morency, L. P. (2018). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2), 423–443.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 1597–1607.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. (2020). Big self-supervised models are strong semi-supervised learners. In *Advances in Neural Information Processing Systems (NeurIPS 33)*.

Desai, S., and Johnson, J. (2021). VirTex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11162–11173.

Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pp. 4171–4186.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. (2021). An image is worth 16×16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9729–9738.

Jia, C., Yang, Y., Xia, Y., et al. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*.

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. Technical report, University of Toronto.

Radford, A., Kim, J. W., Hallacy, C., et al. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*.

Schuhmann, C., Vencu, R., Beaumont, R., et al. (2021). LAION-400M: Open large-scale image-text data. *arXiv preprint arXiv:2111.02114*.

Sharma, P., Ding, N., Goodman, S., and Soricut, R. (2018). Conceptual captions: A cleaned, large-scale image captioning dataset. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Zhang, R., Isola, P., and Efros, A. A. (2016). Colorful image colorization. In *European Conference on Computer Vision (ECCV)*, pp. 649–666.