CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Hybrid-Technical and Behavioral Attack Attribution in Cyber Threat Intelligence

by

Ehtsham Irshad

A dissertation submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Computing
Department of Computer Science

2024

Hybrid-Technical and Behavioral Attack Attribution in Cyber-Threat Intelligence

By

Ehtsham Irshad

(DCS183005)

Dr. Raheel Nawaz, Professor

Staffordshire University, UK

(Foreign Evaluator 1)

Dr. Zeeshan Pervez, Professor

University of Wolverhampton, UK

(Foreign Evaluator 2)

Dr. Abdul Basit Siddiqui

(Research Supervisor)

Dr. Abdul Basit Siddiqui

(Head, Department of Computer Science)

Dr. Muhammad Abdul Qadir

(Dean, Faculty of Computing)

DEPARTMENT OF COMPUTER SCIENCE

CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY

ISLAMABAD

2024

Dedicated to my parents and family, whose encouragement and support enabled me to reach this goal

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD

Expressway, Kahuta Road, Zone-V, Islamabad
Phone: +92-51-111-555-666 Fax: +92-51-4486705
Email: info@cust.edu.pk Website: https://www.cust.edu.pk

CERTIFICATE OF APPROVAL

This is to certify that the research work presented in the dissertation, entitled "**Hybrid-Technical and Behavioral Attack Attribution in Cyber Threat Intelligence**" was conducted under the supervision of **Dr. Abdul Basit Siddiqui**. No part of this dissertation has been submitted anywhere else for any other degree. This dissertation is submitted to the **Department of Computer Science, Capital University of Science and Technology** in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the field of **Computer Science.** The open defence of the dissertation was conducted on **November 11, 2024**.

Student Name: **Ehtsham Irshad** (DCS183005)

The Examination Committee unanimously agrees to award PhD degree in the mentioned field.

Examination Committee:

(a) External Examiner 1: Dr. Zunera Jalil
Professor
Air University, Islamabad

(b) External Examiner 2: Dr. Hassan Mujtaba
Professor
FAST-NU, Islamabad

(c) Internal Examiner: Dr. Mohammad Masroor Ahmed
Professor
CUST, Islamabad

Supervisor Name: Dr. Abdul Basit Siddiqui
Associate Professor
CUST, Islamabad

Name of HoD : Dr. Abdul Basit Siddiqui
Associate Professor
CUST, Islamabad

Name of Dean: Dr. Muhammad Abdul Qadir
Professor
CUST, Islamabad

AUTHOR'S DECLARATION

I, **Ehtsham Irshad (Registration No. DCS183005)**, hereby state that my dissertation titled, '**Hybrid-Technical and Behavioral Attack Attribution in Cyber Threat Intelligence**' is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/ world.

At any time, if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my PhD Degree.

(**Ehtsham Irshad**)

Dated: 11, November, 2024

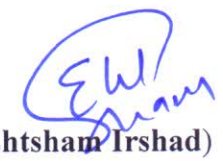
Registration No: DCS183005

PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the dissertation titled "**Hybrid-Technical and Behavioral Attack Attribution in Cyber Threat Intelligence**" is solely my research work with no significant contribution from any other person. Small contribution/ help wherever taken has been duly acknowledged and that complete dissertation has been written by me.

I understand the zero-tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled dissertation declare that no portion of my dissertation has been plagiarized and any material used as reference is properly referred/ cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled dissertation even after award of PhD Degree, the University reserves the right to withdraw/ revoke my PhD degree and that HEC and the University have the right to publish my name on the HEC/ University Website on which names of students are placed who submitted plagiarized dissertation.



(**Ehtsham Irshad**)

Dated: 11, November, 2024

Registration No: DCS183005

List of Publications

It is certified that following publication(s) have been made out of the research work that has been carried out for this dissertation:-

1. **E. Irshad**, A. B. Siddiqui, "Cyber threat attribution using unstructured reports in cyber threat intelligence", *Egyptian Informatics Journal*, vol. 24, issue 1, pp. 43-59, 2023.

2. **E. Irshad**, A. B. Siddiqui, "Context-aware cyber-threat attribution based on hybrid features", vol. 10, issue 3, pp. 553-569, *ICT Express*, 2024.

3. **E. Irshad**, A. B. Siddiqui, "Using machine learning techniques for accurate attack detection in intrusion detection systems using cyber threat intelligence feeds",*IJCSNS International Journal of Computer Science and Network Security*, vol.24 no.4, pp. 179-191, 2024.

(Ehtsham Irshad)
Registration No: DCS183005

Acknowledgement

All thanks to Allah Almighty, who has all the knowledge and the power to bestow peace, mercy, and blessing on his final messenger. Sincere thanks to my supervisor, Dr. Abdul Basit Siddiqui, whose continual encouragement and guidance enabled me to complete my dissertation. Despite his numerous duties and tasks, he continuously made himself accessible to supplement my erroneous ideas with his great skills and solid technical background, for which I am eternally thankful. This dissertation would not have been completed without his counsel and assistance.

Furthermore, I am grateful to my guidance committee members and the dean of faculty of computing, Dr. Muhammad Abdul Qadir, for their supervision, support, and guidance.

The love, prayers, and support of my parents, wife, and children, who effectively shared my obligations and handled all home issues independently. It allowed me to stay focused on my studies, allowing me to finish this dissertation. I am grateful to my colleagues Dr. Muhammad Rizwan and Dr. Sohail Sarwar for their insightful advice. Aside from those stated above, many others assisted me in reaching this position. May Allah bless them all and bring them peace and prosperity. I am also grateful to everyone who prayed for me during my PhD.

(Ehtsham Irshad)

Abstract

The process of identifying the perpetrators accountable for a cyber-attack is known as cyber-attack attribution. This is a difficult undertaking since attackers conceal their identities using various obfuscation and deception techniques. A digital forensic investigation is carried out following an attack to collect data from network/system logs. After the investigation is completed, the report is published in a variety of formats including text and PDF. Due to the lack of standardized publishing procedures, extracting valuable information from these reports is a difficult undertaking. Manual feature extraction from unstructured cyber threat intelligence (CTI) reports is a challenging task. An automated mechanism is needed to extract features for threat mapping.

The goal of this research dissertation is to develop a mechanism for attributing cyber-threat actors (CTAs) using hybrid features (technical and behavioral) by using machine/deep learning techniques. Previously, this mapping was only carried out by extracting limited features, i.e., tactics techniques and procedures (TTPs), tools, and malware. Characteristics such as target country, organization, and application have not been exploited in the research so far.

The features used in this domain to date do not provide any information about the behavioral characteristics of the attacker, i.e., objectives, motivations, and goals. Contextual attacker profiles are required due to the rapid growth of technology and the continual change of attacker tools and strategies. It is critical to incorporate behavioral characteristics into the cyber-threat attribution process to understand the actors and their environment.

This research dissertation proposes an innovative concept of incorporating hybrid features into the cyber-attack attribution process. A novel model "attack2vec" trained for domain-specific embedding has been proposed for feature extraction. The results of the novel model have been compared with various baseline methods. Features have been validated against benchmark frameworks such as MITRE ATT&CK, and threat actor encyclopedia. Optimal characteristics have been chosen for CTA. Performance metrics that are used in this study include Accuracy, Precision, Recall, and F1-measure. Following experimentation, various

machine/deep learning algorithms were employed to achieve 97%, 98%, 98%, and 97% for Accuracy, Precision, Recall, and F1-measure by using the novel embedding model. These include long short-term memory (LSTM), decision tree, random forest, and support vector machine algorithms.

The function of CTI is to deliver advanced feeds for precise attack detection in IDS. The role of CTI feeds for IDS is also examined in this research dissertation. Various datasets have been analyzed. With the proposed study, machine learning algorithms have improved the ability to recognize network attacks. The proposed model produces 98% Accuracy, 97% Precision, and 96% Recall.

Contents

List of Figures

List of Tables

Abbreviations

ANN	Artificial neural network
APT	Advanced persistent threat
ARM	Association rule mining
BERT	Bidirectional encoder representations from transformers
CAPEC	Common attack pattern enumeration and classification
CBOW	Continuous bag of word
CISA	Cyber-security & infrastructure security agency
CKC	Cyber-kill chain
CNN	Convolutional neural network
CTA	Cyber-threat actor
CTC	Consolidated tree construction
CTI	Cyber-threat intelligence
DeLP	Defeasible logic programming
DoS	Denial of service
DLTIF	Deep learning threat-intelligence identification framework
DLNN	Deep learning neural network
ETDA	Electronic transactions development agency
GA	Genetic algorithm
HIDS	Host Based intrusion detection system
IDS	Intrusion detection system
IoC	Incident of compromise
IL-CyTIS	Integrated lightweight cyber threat information structure
IoT	Internet of things

IntruD-Tree	Intrusion trees
IPS	Intrusion prevention system
LLM	Large language model
LSI	Latent semantic indexing
LTSM	Long short term memory
ML	Machine learning
NLP	Natural language processing
OSIF	Security open source intelligence framework
PCA	Principal component analysis
ROC	Receiver operating characteristic curve
RF	Random forest
RNN	Recurrent neural network
SIMVER	Similarity based vector representation
SIEM	Security information and event management
SMOBI	Smoothed binary vector
SOC	Security operation center
STIX	Structured threat information expression
TAL	Threat agent library
TF-IDF	Term frequency inverse document frequency
TTP	Tactics techniques and procedures
SVM	Support vector machine

Symbols

Entropy(S)	It is the dataset or subset of data at a particular node
c	Number of classes or categories in the dataset
p_i	Proportion of the number of elements in class i to the total number of elements in S
$\|Sv\|$	Number of instances in S for which attribute A has value v
S	is the total number of instances in the dataset
X_i	It is standarized feature
$\bar{X}_i$	It is the mean
i_t	Input gate
f_t	Forget gate
c_t	Cell state
o_t	Output gate
σ	Sigma
$\sum$	Summation

Chapter 1

Introduction

In today's interconnected society, cyber-security is critical. It is used for protecting systems, networks, and data from emerging attacks. A data breach can have serious consequences for an organization's reputation. In this chapter importance of cyber-security, and the role of AI in this domain is elaborated in detail. The importance of CTI and its life cycle is also discussed.

1.1 Importance of Cyber-Security

Customer trust is difficult to gain and easy to lose. Effective safeguards can assist in preserving customer, partner, and stakeholder trust. Critical infrastructure is strongly reliant on interconnected systems [1–3]. In an increasingly digital world, it is a proactive approach to mitigate risks and protecting against potential dangers [4, 5].

Protecting sensitive data is the main reason businesses invest in cyber-security. The areas include private intellectual property, financial data, and customer information. Data breaches caused by cyber-attacks can have serious repercussions, including monetary losses and reputational harm. Businesses that put cyber-security as a priority are in a better position to maintain an advantage over rivals in the marketplace. Businesses can obtain a competitive advantage by focusing on

their core skills and decreasing the risk of cyber threats by implementing effective security measures [6–9].

New risks are introduced by businesses as they implement new IT solutions and technology. As cybercrime becomes more professionalized, there are more diverse and advanced risks. The most cutting-edge cyber-security systems are continually being circumvented or surpassed by CTAs. These systems conceive, develop, and update regularly. The current setup like hardware and software firewalls, data encryption strategy, and user authentication techniques is not enough to cater the modern-day attacks. Unfortunately, this equipment is not able to secure the computer networks from cyber-attacks [10–12]. AI plays an important role in cyber-security, providing increased capabilities in threat detection, response, and mitigation.

1.2 Role of Artificial Intelligence (AI) in Cyber-Security

AI can analyze vast amounts of data for patterns and anomalies that could identify potential cyber threats. Traditional methods are outperformed by machine learning models that can learn from previous data to detect expected and unexpected dangers. AI can monitor user and system behaviors to build baselines and detect deviations that may indicate potential breaches or unauthorized activity [13, 14].

AI-powered technologies, for example, can immediately isolate compromised systems or prevent unauthorized access, reducing the impact of an attack. The role of AI in this field is increasing with time and it is widely used in every industry. By analyzing current and past data, AI can foresee possible security concerns, allowing organizations to resolve vulnerabilities before they are exploited [15, 16]. Cyber-security and AI have been heralded as transformation technologies that are far closer than we realize. The role of AI in cyber-security is evolving with more sophisticated tools and strategies available to battle an increasingly complex and dynamic threat landscape. As we consider the potential security implications of

machine learning and AI, it is necessary to first understand the current cyber-security issues [17–20].

1.3 Cyber-Threat Intelligence (CTI)

CTI is a knowledge base that encompasses context, behavior, actions, and the results of an assault. This information is gathered, analyzed, and applied to better understand cyber threats and vulnerabilities. This information is then analyzed to discover potential cyber risks [21–28]. This knowledge base is made available by CTI to help organizations defend against cyber-attacks. It also gives them the chance to access challenges and threats they are currently facing online and to make informed decisions about potential future attacks. The purpose of CTI is to proactively detect and reduce potential cyber threats. It assists organizations in understanding the tools employed by cyber attackers, allowing them to anticipate and protect against prospective threats [29–31].

It is an emerging field that is crucial to modern cyber-security tactics. As cyber-threats become more complex and widespread, the demand for rapid, accurate, and actionable intelligence has never been greater. CTI gives a complete picture of the threat landscape by combining technical data, like as malware signatures and network traffic patterns, with behavioral insights, such as attacker motivations and strategies. This comprehensive strategy not only improves an organization's ability to defend against current threats, but also prepares it for future issues, making CTI an essential component of proactive cyber-security initiatives. It allows for a more proactive and informed approach to cyber-security, which aids in the protection of networks, systems, and sensitive data.

Database repositories for individual nations are currently being created [32–34]. It has been advantageous to thwart potential attacks by supplying this database with security equipment. Today, businesses are concentrating on creating their knowledge bases using publicly available data. Based on this information, threat feeds are created in standard structured threat information expression (STIX) format [35–37].

1.4 Cyber-Threat Intelligence Life Cycle

CTI life cycle consists of six phases as illustrated in figure 1.1. It is critical for systematically controlling and mitigating cyber risks. It starts with planning and direction, which focuses on identifying objectives and connecting intelligence collection with organizational requirements. This phase ensures that efforts are focused on important threats by establishing clear priorities and determining the information required, like threat actors or attack pathways. Alignment with security strategies is critical for effective resource allocation. Following planning, the collection phase entails acquiring raw data and information from a variety of sources. This includes identifying sources like open-source intelligence, threat feeds, and internal logs, as well as gathering data related to identified threats, such as indications of compromise (IOCs) and threat actor strategies. This stage is critical because it supplies the raw material required for later analysis.

The processing phase involves transforming the acquired data into a structured and useful format. It is important process here since they ensure that the data is correct, comprehensive, and consistent. This preparation is critical for effective analysis, since it converts raw data into a more digestible and cohesive format. In the analysis step, the processed data is reviewed to extract useful insights and actionable intelligence. This entails analyzing data, applying threat modeling frameworks, and evaluating risks. The purpose is to comprehend the nature of risks, detect patterns, and assess probable consequences, allowing for more educated threat mitigation decisions.

Once the analysis is completed, the results are communicated during the dissemination phase. Detailed reports, dashboards, or briefings are prepared and distributed to appropriate stakeholders, such as incident response teams or executives. Effective dissemination ensures that actionable intelligence reaches those who require it to make informed decisions and take appropriate action. Finally, the feedback and review phase entails assessing the effectiveness of the intelligence and fine-tuning the CTI process based on performance assessments and feedback. This constant assessment enables changes to methodologies, tools, or sources, ensuring that intelligence remains relevant and effective as threats emerge [38].

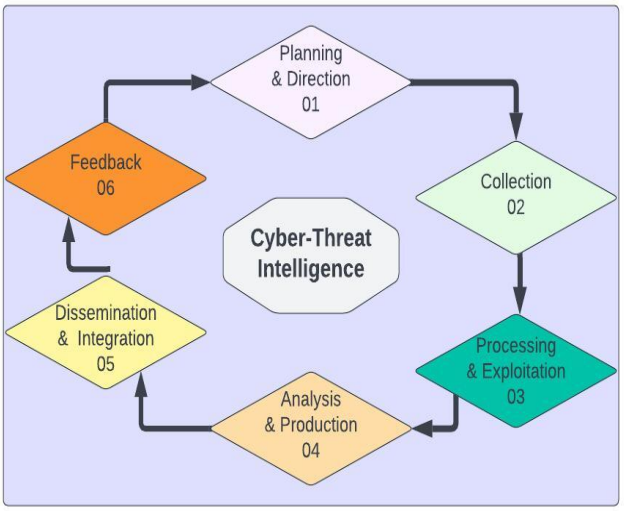


FIGURE 1.1: CTI Life Cycle Processes.

1.5 Types of Cyber-Threat Intelligence

There are several types of CTI including tactical, operational, strategic, and technical as shown in figure 1.2. High-level management uses strategic CTI. This form of intelligence provides a high-level view of long-term cyber trends and dangers. It assists organizations in making useful resource allocation, policy formulation, and overall security strategy decisions. Tactical CTI may provide information on TTP, malware, and attacker tools. It is advantageous for IT managers and SOC analysts. It is primarily concerned with the immediate future and assists organizations in understanding existing threats and vulnerabilities. It offers security teams with actionable information to respond against imminent threats.

Operational CTI provides information on incoming attacks. It gives insight into a specific IoC and is beneficial to SOC employees. It is more detailed than strategic or tactical intelligence and assists organizations in understanding threat actors TTPs. It can reveal how an attack is carried out and how to protect against it. Technical intelligence is concerned with the exact technological aspects of cyber threats. This intelligence is essential for identifying and mitigating specific hazards. [39–42].

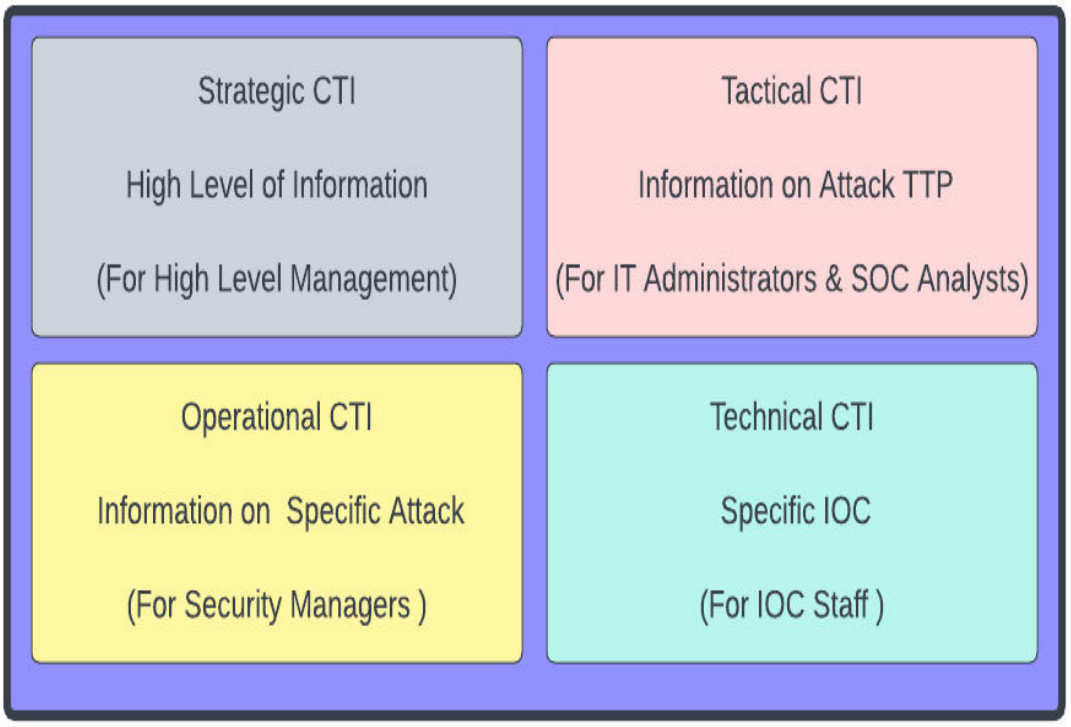


FIGURE 1.2: Types of CTI.

1.6 Importance and Challenges of CTI

CTI aids in the identification and evaluation of potential threats to an organization's infrastructure, systems, and data. Understanding these risks aids in the effective implementation of mitigation measures. This includes the benefit of identifying and reducing possible risks before they cause harm. This also saves from averted security breaches. By integrating high-quality threat information with additional technologies, threat-detection, and defense capabilities can be greatly increased [43].

Security teams can effectively respond to threats and lessen the impact of an attack by using this information. It allows organizations to proactively fight against possible cyber-threats by recognizing and comprehending threat actors techniques. Organizations can better plan and build their defenses by staying ahead of potential threats [44, 45].

1.7 Cyber-Attacks Statistics

In today's world, cyber-attacks are growing very quickly. There are assaults on almost every industry. The percentage of the organization that has been attacked at least once is shown in figure 1.3. It shows that attacks have increased with time and in 2021 86.2% of organizations were attacked at least one time. An estimate of a data breach cost is shown in figure 1.4. The cost of data breaches is also increasing very rapidly. According to this graph, cybercrime will cost an estimated 23.82 trillion US dollars in 2027 [46, 47]. As a result, organizations are suffering significant losses as the percentage of attacks rises. Thus, to safeguard themselves against cyber-crime, organizations must develop defense mechanisms.

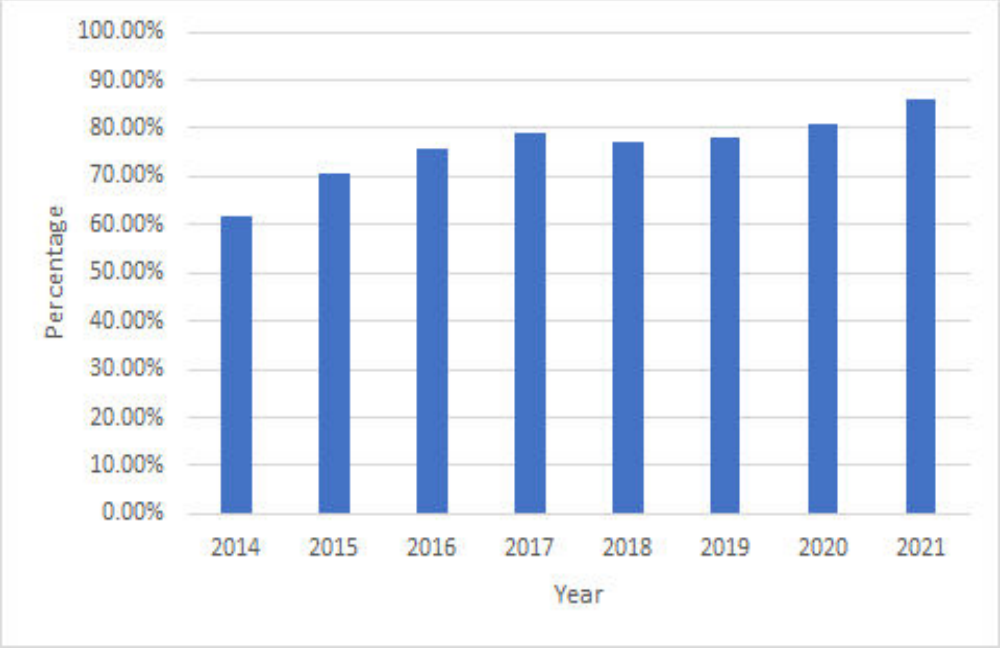


FIGURE 1.3: Statistics Cyber-Attacks on Organizations.

The amount of information available can be challenging. It is difficult to filter through massive amounts of data to uncover useful and actionable intelligence. It is critical to ensure that the acquired data is accurate, relevant, and reliable. Incorrect assessments and responses might result from false or misleading information. Cyber attacker TTPs are always evolving. It is a constant challenge to stay ahead of these sophisticated threat actors and understand their developing

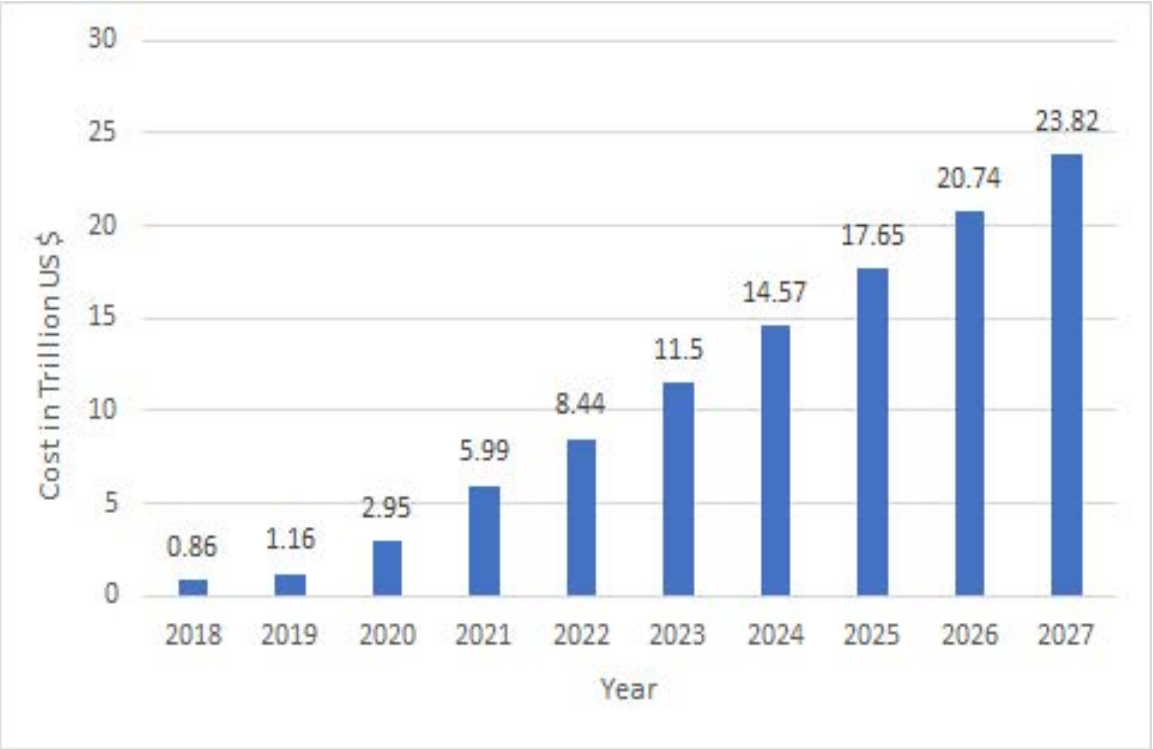


FIGURE 1.4: Estimated Cost of Cyber-crime in Different Years.

techniques [48–50]. One of the biggest challenges of CTI is to extract useful information from vast amounts of raw data [51–53].

There are numerous profiles of attackers. From the attack patterns depicted in CTI reports, security analysts attempt to identify the perpetrators. Numerous well-known companies, including Fire-eye, Trend Micro, Kaspersky, and Symantec release these reports. This data cannot be decoded by machines, but can only be read by knowledgeable security analysts. Numerous manual tasks must be performed to extract useful information from these reports [54, 55].

1.7.1 Case Study

A major cyber-security breach at National Database and Registration Authority (NADRA) in March 2023 exposed national information. Several weaknesses in NADRA systems were discovered by a Joint Investigation Team (JIT), which was established in October 2023. A thorough investigation was conducted by law enforcement, cyber security specialists, and experts from the Federal Investigation

Agency (FIA) as a result of the intrusion, which sparked worries about data security and integrity.

The JIT report, which was released in early 2024, identified particular security flaws and the people in charge of the oversight errors that made the assault possible. Additionally, it suggested criminal and disciplinary measures, such as firing specific NADRA officers. The government responded by ordering NADRA to modernize its systems, put in place stronger cyber-security safeguards, and move quickly to stop future occurrences of this kind.

This data breach incident exposed the personal data of around 2.7 million persons. Vulnerabilities in NADRA security system, particularly at its offices in Karachi, Multan, and Peshawar, where officials were allegedly involved in leaking sensitive data, was the main cause of the breach. It was discovered that this data has been exported to Romania and Argentina. Although the precise tactics employed in the attack are not entirely known, earlier studies identified supply chain attacks and biometric system defects that might have played a role in the data leak. In addition to suggesting technology improvements, the JIT suggested disciplinary and criminal measures against the offending parties.

1.8 Data Sources of Cyber-Threat Intelligence

Since CTI has access to a large amount of data, separating the relevant information can be difficult. Identifying the data sources for CTI is crucial. While they give important intelligence, the quality and significance of the information gathered will be determined by the context, dependability of the sources, and unique needs of the organization and individual seeking this intelligence. In this research investigation, unstructured CTI reports have been used for extracting technical features. The following are available data sources [56, 57].

- CTI feeds (STIX format)
- Open source intelligence (OSNIT)

- Network/server logs
- Hacker forums
- Social media (twitter, facebook etc.)
- Honeypots
- Unstructured CTI reports
- Common vulnerabilities and exposures (CVE)
- National vulnerability database (NVD)
- Blogs
- Threat advisories
- Security websites
- Dark web
- Repositories e.g., GitHub

1.9 Benchmark Frameworks in CTI

The most widely used framework in this domain is MITRE ATT&CK [58–62]. According to table 1.1 (accessed August 24, 2024), this database provides information on threat actor's tactics, techniques, software, mitigation, and data sources. There are frequent updates to this database. There are 152 cyber-threat actors, 202 techniques, 435 sub-techniques, 14 tactics, 794 software, 43 mitigation techniques, and 41 data sources in this framework.

A model that illustrates the seven phases of the attack process is the cyber-kill chain (CKC). It was one of the most extensively used models prior to MITRE, It lays out the stages of a cyber attack, providing an organized approach to analyzing security systems and defense measures at each stage. Within this framework,

TABLE 1.1: Number of Features of MITRE Framework.

Sr. #	Features	Quantity
1.	Groups	152
2.	Techniques	202
3.	Sub-techniques	435
4.	Tactics	14
5.	Software	794
6.	Mitigation	43
7.	Data sources	41

there are seven steps: reconnaissance, weaponization, delivery, exploitation, installation, command & control, and actions on objectives. The cycle starts with reconnaissance, in which attackers gather extensive knowledge about their target, such as network structures and potential vulnerabilities, frequently using passive methods like social engineering or scanning. This is followed by the weaponization step, in which attackers develop a specialized exploit, such as malware or phishing emails, to exploit the detected vulnerabilities. During the distribution step, the weaponized exploit is sent to the victim via a variety of vectors, including email attachments and malicious websites. Once provided, the exploit code is executed, allowing the attacker to acquire initial access by leveraging the vulnerability.
Installation phase follows, in which the attacker gets a foothold in the target system by installing malware or other tools to assure long-term access. During command and control (C2), the attacker establishes a communication channel to manage and control the hacked system, frequently employing covert tactics to evade discovery. Finally, during the acts on objectives phase, the attacker focuses on their core aims, which may involve data exfiltration, system disruption, or deeper infiltration. Understanding each level of the Cyber Kill Chain enables organizations to create tailored defenses and monitoring techniques to successfully detect and mitigate threats at each stage [63, 64].
This model is important in CTI because it provides a structured framework for

analyzing and comprehending the course of cyber attacks. The CKC model allows organizations to identify and manage vulnerabilities at each stage of an attack life cycle by breaking it down into different phases. Its significance stems from its ability to enable proactive defense strategies. The CKC model has several advantages, including increased situational awareness, which aids in mapping out attack strategies and approaches, improved incident response through a better knowledge of assault patterns, and more effective resource allocation by focusing on important phases of the attack.

Another framework used in this domain is CAPEC [65–67]. Each framework offers unique benefits and focuses on different aspects of CTI. Organizations commonly combine these frameworks to establish a solid threat intelligence program based on their specific needs and objectives.

1.10 Time is a Critical Factor in Cyber-Threat Intelligence

Cyber-threat knowledge aids in the early detection of threats and attacks. The sooner a company detects an assault, the more time it has to respond, mitigate, and minimize potential damage. Timely information ensures a quick response to prevent and lessen the impact of an assault as depicted in figure 1.5. Timely detection enables security teams to adjust and improve their defense plans in real-time to effectively tackle new attacks. Access to real-time threat intelligence aids in the detection of prospective threats before they manifest as full-fledged attacks. This information enables organizations to put preventative measures in place, lowering the chance of successful breaches.

Time is an important component in this field. An attack has less effect if it is detected early. If it is discovered later, the consequences are substantial. The cost of a breach is estimated at $4.35 million, it takes the industry 197 days to detect and 69 days to contain[68].

The value of time resides in its ability to deliver timely, relevant, and actionable

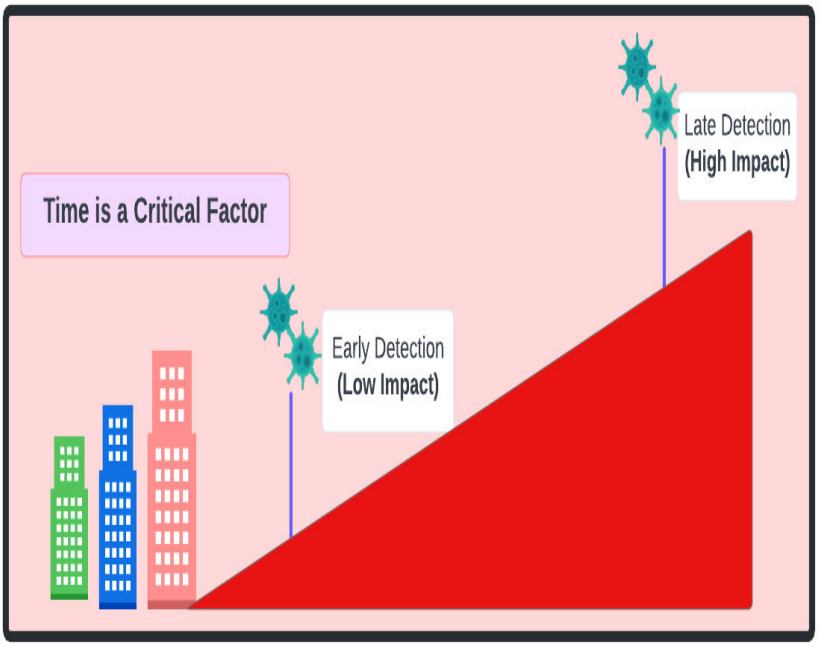


FIGURE 1.5: Impact of Time in Attack Detection.

information to boost an organization's defense against fast-growing and sophisticated cyber threats. The faster an organization gathers, analyzes, and acts on threat intelligence, the better it is ready to defend [69, 70].

1.11 Advanced Persistent Threats (APT)

APT are sophisticated, ongoing, and targeted cyber-attacks carried out by determined adversaries, i.e., nation-states, organized criminal organizations, or other well-resourced entities. These are distinguished by their stealth, long-term nature, and narrowly focused objectives. APT actors are tenacious in their attempts, employing a deliberate and continuous strategy to infiltrate a target network or system. They intend to remain unnoticed for as long as possible to fulfill their objectives. These attacks usually have numerous stages and can last for a long time. These types of attacks go unnoticed for longer periods. As APT is state-sponsored therefore attribution is difficult [71–74]. Because of the complexity and endurance, protecting against them necessitates a multi-layered security approach

that includes strong cyber-security practices, constant monitoring, and user education.

1.12 Identification of Attack Steps and Patterns

The ability to recognize the attack steps used by attackers, as depicted in figure 1.6, is a significant challenge in this field. These actions describe the attacker's entire attack flow. The CKC model specifies these steps. This indicates which technique or threads the attacker is using at which step [75–77].

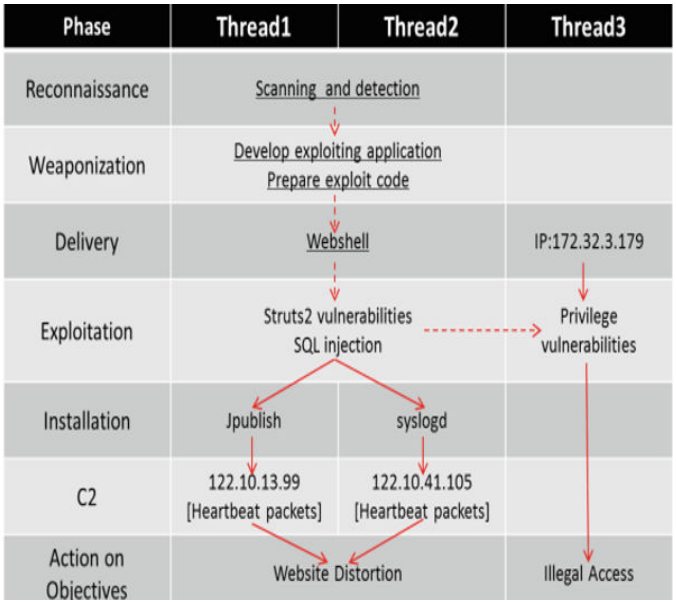


FIGURE 1.6: Steps used by Attackers.

Finding the patterns and regularities is a difficult task. It aids in the identification of the attacker. It will assist in establishing the connection between the attack's various characteristics. [78–81]. It can reveal details about who, what, where, why, and how the attack occurred.

As seen in figure 1.7, these five parameters provide information about the attack's strategy, implementation, and methodology. The who parameter reveals the real individual, group, or nation responsible for an attack. The identification of an attack's true adversary is a crucial step. To lessen the impact of upcoming assaults, this information may be helpful. What parameter indicates the attack's general

scope? It explains the attacker's goals for this attack. The direction is indicated by the where parameter. It indicates the exact time of an attack. The attacker's aims and goals are revealed by the why parameter. How indicates the instruments and methods the actor is using [82].

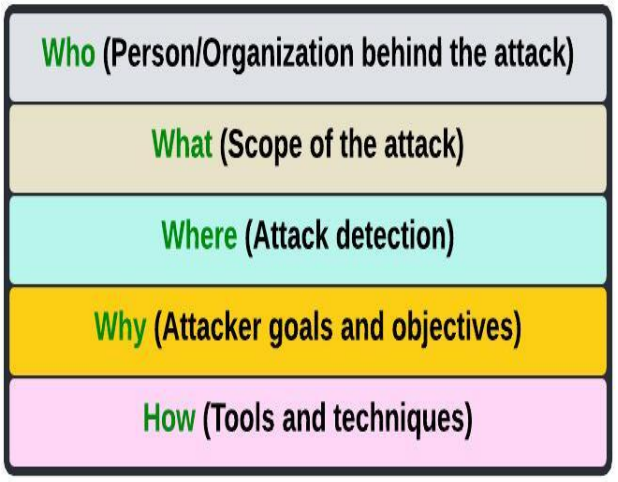


FIGURE 1.7: Parameters of a Cyber-Attack.

Understanding the TTP that the attacker is using can help counteract future attacks [83]. Cyber-threat attribution is difficult since attackers typically use a variety of techniques to hide their identity. Organizations can anticipate future threats and effectively take preventive action against these attacks.

1.13 Pyramid of Pain Model

The pyramid of pain model [84–89] is depicted in figure 1.8. IOC comes in two varieties, low and high level. Domain names, source/destination ports, hashes, and IP addresses are examples of low-level. Due to their transience, these traits are easily altered by the attacker. Malware, tools, and TTP are high-level IOCs. These characteristics are permanent, have a big impact, and are difficult for an attacker to alter.

Low-level IoCs like IP addresses, file hashes, or domain names are not commonly

employed for cyber-threat attribution as they can be easily altered. These indications are unreliable for long-term attribution because threat actors can change them to avoid identity. Additionally, they lack the larger context required to comprehend the goals or operational strategies of an attacker. On the other hand, high-level IoCs like tactics, methods, and procedures (TTPs) offer more reliable and consistent behavioral patterns that are more difficult to alter without affecting the attacker's overall plan. Because they reveal more complex elements of an adversary's identity and operational objectives, high-level IoCs are more useful for attribution and enable a more accurate and dependable threat actor identification. High-level IoCs are a better fit for thorough and long-term threat actor attribution because of frameworks like MITRE ATT&CK, which track adversary activities.

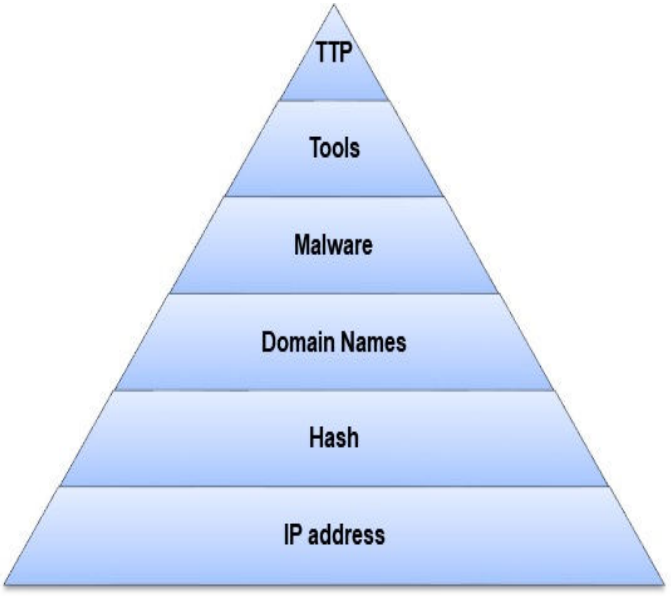


FIGURE 1.8: Pyramid of Pain Model.

1.14 Role of Security Devices in CTI

Security devices play a crucial role in the context of CTI. These devices act as both sources and consumers of threat intelligence, assisting in the defense against known and developing threats. Firewalls, IDS/IPS systems, antivirus software, and secure web gateways are all helpful in detecting and preventing various cyber

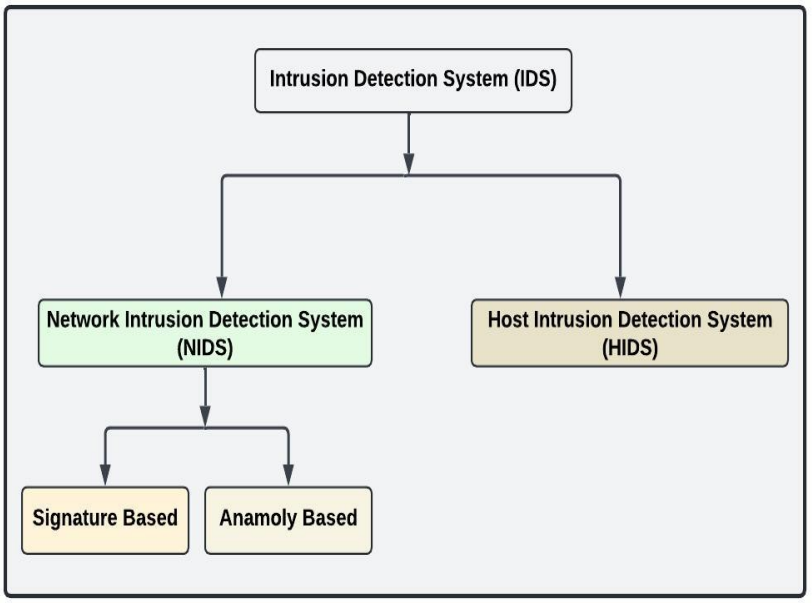


FIGURE 1.9: Types of IDS.

threats. They employ threat-intelligence feeds to identify known malware signatures, suspicious activities, or behaviors that may suggest an impending attack. These devices serve as the first line of defense in detecting and stopping harmful activity. Threat intelligence feeds from many sources, including various platforms, suppliers, open-source feeds, and internal sources are integrated into these devices [90, 91].

To secure networks, devices like firewalls and IDS/IPS are employed. Businesses can be destroyed by false positive alarms, which happen frequently, and by failing to recognize zero-day attacks. To find malicious traffic in the network, IDS performs deep packet inspection. Every traffic passing through it is scrutinized and signature databases are used to compare the payload. If a match is found, the network permits the request to proceed, otherwise, it is blocked [92, 93]. IDS are of two IDS types as depicted in figure 1.9. NIDS is used for network-based activity; a HIDS is installed on the host to detect attacks. There are two types of NIDS. Signature-based is one type. It maintains a repository of all known attack signatures, so when a request comes in, it first compares to the signature database. If the request matches, it is denied or rejected. The second type of detection is based on anomalies or behavior. This type is used to find zero-day attacks [94–96].

The process of finding patterns in data that deviate from predetermined normal behavior is known as anomaly detection [97–100].

1.15 Role of AI in CTI and IDS

AI contributes significantly to CTI by improving data gathering, analysis, and reaction to cyber threats. It aids in the processing and filtering of threat intelligence feeds from many sources, providing security professionals with relevant and actionable data. The ever-improving computational capabilities of digital systems, together with upgraded TTPs employed by cyber criminals, do not match the conventional security mechanism for detecting intrusion and mitigating threats in today's cyber-security environment [101, 102].

It is important to highlight that human skill is still required. Because AI systems may not always understand the greater commercial context or societal consequences behind risks, human analysts are required to assess results. Finally combining AI and human intelligence in CTI is becoming increasingly crucial to effectively battle the ever-changing panorama of cyber-threats. [103–106].

Accurate defense measures in the form of ML-based IDS are required to protect against cyber-attacks. They are being deployed as potential solutions for identifying network attackers [107–109]. AI-powered IDS can handle big and complicated infrastructures. There is a need for a method to categorize network assaults because the IDS continues to struggle more accurately with improving detection accuracy despite the significant research efforts [110, 111]. Several datasets have been used to evaluate machine learning algorithm's performance.

1.16 Role of CTI in IDS

CTI regularly updates information and context about cyber-attacks. It provides multi-source databases that assist cyber defense mechanisms, enabling thorough monitoring, detection, and reaction to online threats. Without peering beyond

the network of your organization, it can be challenging to predict when and how an attack will occur. Using global CTI feeds will give information about how an attack is happening and who is behind it. Threat feeds will help to develop important defensive security strategies. It empowers analysts to decide how to respond against impending attacks [112].

For generating threat feeds, CTI uses internal community and outside sources. Data gathered from corporate security solutions like IDS/IPS, firewalls, and antivirus software among others is included in internal feeds. An example of an external source is a threat feed from a public or a paid private source from several well-respected and reputable security vendors.

Today, information sharing across organizations in the relevant business is difficult. Many security organizations (Alien Vault, threat connect, etc.) offer CTI feeds. They are being integrated into devices. These feeds are constantly updated.

1.17 Dissertation Outline

Chapter 2

This chapter provides a detailed definition of cyber-attack attribution. CTA methodologies and overall architecture are explained. Various metrics, statistics, and sources of CTI are expanded. Furthermore, the research motivation, goal and objectives, problem statement, gaps, research questions, and methodology are presented.

Chapter 3

A comprehensive literature review is conducted in this chapter for CTA attribution and accurate detection of attacks in IDS. Critical analysis and challenges of research work are explained in detail. Finally, important findings of the literature review are elaborated.

Chapter 4

This chapter provides a detailed explanation of the planned methodology for this research. The methodology for obtaining technical, behavioral, and hybrid aspects

is explained. The optimal feature selection strategy is explained. The research dissertation's experimental approach is explained near the end.

Chapter 5

This chapter outlines the experiment and the outcomes. Results for technical, behavioral, and hybrid features are presented. Finally, the findings and tests for optimal feature selection are complete.

Chapter 6

This chapter goes into detail about the research's conclusion. Furthermore, future work is described.

Chapter 2

Background

In this chapter cyber-attack attribution, its levels, techniques used in this domain, architecture, performance metrics, types of features, and dataset used are discussed in detail. Also, the motivation for the research, research aim/objectives, problem statement, research challenges and gaps, problem statement, research questions, methodology, and contributions are discussed.

2.1 Cyber-Attack Attribution

Cyber-attack attribution is to know about the person or organization behind an attack. It aims to pinpoint who is accountable for online activities. Similar to actual violence, attribution involves both technical and political evaluations. Technical techniques include virus analysis and scripts that link online influence activities to well-known individuals. Political approaches are intimately related to intelligence gathering, analysis, and the political choices that influence whether or not to attribute operations publicly [113, 114].

There are different profiles and various attributes of the attacker. It is a challenging task to attribute cyber-threat actors based on attack patterns extracted from unstructured CTI reports. To develop informed decisions regarding the origin and identity of cyber attackers, attribution is a difficult task that frequently involves

a combination of behavioral analysis, technological forensics, information sharing, geopolitical context, and other elements[115].

2.2 Levels of Attribution

Figure 2.1 illustrates the various levels of attribution. The first step is knowing the attacker's tools, TTP. Understanding the nation that carried out the attack is second level. It explains the purpose and motivation for the attack. Understanding the perpetrator of the attack is the third and most crucial level. It is very important to know about the attacker in addition to the attack patterns. This can help the organizations to protect them from future attacks. It would help them to know who is the actual attacker and what is his aim behind it [116–118].

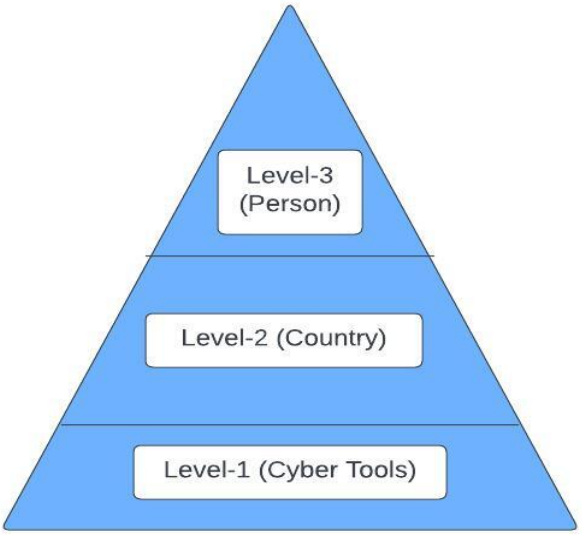


FIGURE 2.1: Levels of Attribution.

2.3 Cyber-Attack Attribution Techniques

Various CTA techniques are shown in figure 2.2. The literature divides attribution techniques into four categories. The first method is based on high-level IOC [119–132]. High-level IOC has long-lasting repercussions, making it challenging

to alter an attacker's tools and techniques. Because they have a great impact, their importance has grown through time. A low-level IOC like IP address, hash, domain name, source/destination port and timestamp is used in the second way of CTA attribution [133–149]. It doesn't have much of an effect because it's so easy for an attacker to change these features. By extracting it, researchers developed several CTA attribution techniques.

The third technique for CTA involves looking for patterns and regularities in the datasets [150–153]. This can help in finding the actual attacker who committed the assault. These strategies use association rule mining to extract relevant patterns from the data. Development of CTA frameworks is the fourth strategy [154–162]. These can provide a framework to find the attack culprit.

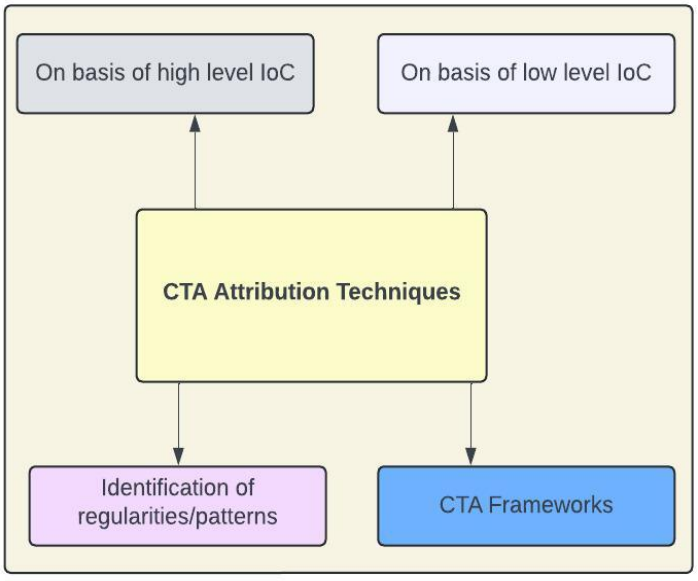


FIGURE 2.2: CTA Attribution Techniques.

2.4 General Architecture for CTA Attribution

Figure 2.3 depicts the general architecture. There are three stages involved in it. The first step is input in which CTI data is collected. The next stage is feature analytics. Text pre-processing is carried out in this stage. The feature

extraction step, which follows text pre-processing employs a variety of state-of-the-art techniques including TF-IDF, word2vec, LSI, and BERT. Following that, features are verified using benchmark framework like MITRE ATT&CK. The next stage, known as classification, involves categorizing the cyber threat actor using a variety of deep/machine learning methods [154, 155, 163].

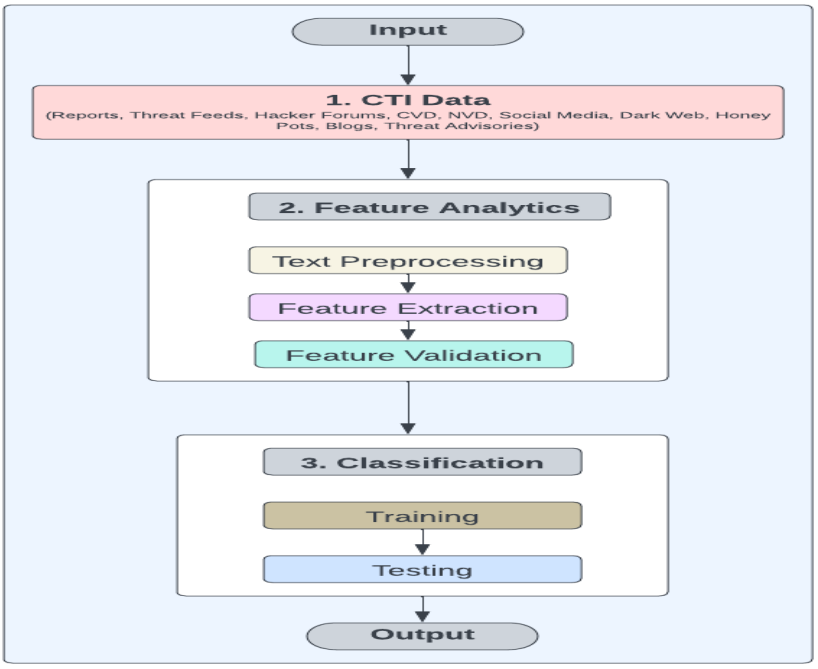


FIGURE 2.3: General Framework for CTA Attribution.

2.5 Performance Metrics

The performance metrics mostly used in this domain are Accuracy, Precision, Recall, and, F1-measure [119–122, 124, 133, 136], confidence, support and lift [150]. Precision is the most used and effective metric used in the literature.

Accuracy is the percentage of correctly classified instances among all. While it is an important criterion for evaluating models, it may not always be sufficient, particularly in imbalanced datasets with uneven distribution of classes. Precision is the ratio of true positive predictions made by the model. It represents the model's ability to prevent false positives. Precision is especially critical in cases where false positives are costly or unwanted.

TABLE 2.1: Datasets used for Cyber-Attack Attribution.

Ref #	Features (Reports)	Year
[136]	17,000	2017
[138]	18,257	2018
[121]	327	2019
[119]	249	2019
[119]	20,630	2019
[121]	Google programmable search engine	2019
[120]	238	2020
[136]	160	2020
[137]	227	2020

Recall calculates the fraction of true positive predictions among all positive instances in the dataset. It represents the model's capacity to catch all positive cases without missing any (reducing false negatives). The recall is critical in situations when all positive cases must be detected, even if it means generating more false positives. The F1 measure represents the harmonic mean of Precision, and Recall. It penalizes models with unbalanced precision and recall levels. Confidence, support, and lift are metrics for finding regularities and patterns in the datasets.

2.6 Datasets for Cyber-Attack Attribution

For extraction of technical features (TTP, tools, malware) unstructured CTI reports are used. They are published in different formats such as PDF and text. Some of the datasets used for cyber threat attribution are shown in table 2.1.

To establish a knowledge base of threat group profiles, Thailand's Computer Emergency Response Team (Thai-cert) has gathered, evaluated, and organized open-source data. The Threat Actor Encyclopedia [164] was used as the dataset

for this research endeavor for extracting behavioral features. It provides a thorough understanding of the motivations and goals of the attack on the part of the threat actor. All of the information in this dataset came from open sources (OSINT). The goal of this encyclopedia was to compile all significant known adversaries of information security. The aim is to increase global threat awareness and aid in quicker crisis response in the future. Information in this dataset is based on data shared by the public security community and does not entirely reflect the views of Thai-cert and ETDA. For maintaining the quality and relevance of this dataset, the key point is data must be kept current and of high quality.

2.7 Types of Features

There are two types of features used in the research. Technical and behavioral features. Attribution is a complex and difficult procedure that frequently necessitates a combination of these characteristics and additional specialized techniques to effectively attribute cyber threats. While these qualities can be useful, attributing cyber threats with absolute confidence can still be challenging and often inconclusive due to the fundamental nature of cyber operations and attackers' ability to conceal their identities.

2.7.1 Technical Features

These include IP addresses, malware signatures, TTP, tools, target country, organization, and application. Technical indicators aid in determining the origin and characteristics of cyber attacks [119–121]. These are discussed below.

TTP:-Understanding the specific methods, strategies, and approaches used by threat actors can provide clues about their identity. This includes examining the tools, procedures, and strategies they employ during an attack. The overarching strategy or purpose of a cyber attack is referred to as tactics. It describes an attacker's higher-level goals. The strategy could be data ex-filtration or espionage.

Techniques are acts employed to carry out the tactics. These may include the deployment of specialized malware, social engineering, vulnerability exploitation, phishing, or other methods used during an attack. Procedures are the step-by-step processes or sequences of actions that threat actors use to carry out an attack. These include the tools utilized, the order of operations, command and control systems, and other specifics about the attack process [58, 79].

It entails identifying patterns, behaviors, and consistent methodology utilized by threat actors throughout multiple attacks. It can give critical insights for cybersecurity professionals, allowing them to anticipate and fight against potential future attacks while also connecting current attacks to known threat actors or organizations.

Malware:- Examining the malware employed in an attack can disclose details about its origins, code structure, and resemblances to previously known malware, providing hints about potential attribution [58, 79].

Tools:- Commercial, open-source, built-in, or publicly available software that a defender, pen tester, red teamer, or attacker could employ. This category includes both software that is not commonly found on enterprise systems and software that is commonly available as part of an operating system that is already present in an environment. PsExec, metasploit and mimikatz are a few examples [58, 79].

Target Country/Organization/Application:- Analyzing these features is critical in this field, as it provides significant information into the source and nature of cyber attacks. When analyzing cyber threats, it is critical to have a thorough grasp of the target country, organization, and application. Understanding geopolitical dynamics and relationships across countries might provide insights into potential motivations for state-sponsored cyber assaults. Certain nations may be embroiled in political tensions or conflicts, which may affect cyber actions against one another. Different industries are targeted for a variety of objectives, including financial gain, espionage, etc. Analyzing the industry can shed light on the motivations behind an attack. Understanding flaws in targeted apps or infrastructure aids in the identification of potential entry points for attackers.

2.7.2 Behavioral Features

The identifiable actions, patterns, or activities displayed by threat actors or hostile entities in the cyber domain are referred to as behavioral traits in cyber-attack attribution. These characteristics are critical for identifying and attributing cyber threats to specific people, organizations, or entities. Table 2.2 displays the behavioral characteristics associated with a CTA that was discovered using the Threat-Agent Library (TAL) [165]. These details provide insight into the actions of CTA. Context-aware profile means identifying the motives and objectives of attackers be-

TABLE 2.2: Behavioral Features by Threat Agent Library.

Sr. #	Features
1.	Attack steps
2.	Success rate
3.	Trace coverage
4.	Outcome
5.	Attack knowledge
6.	Limits
7.	Tools complexity
8.	Resources
9.	Motivation
10.	Actions
11.	Distance to CP
12.	Attacker skill
13.	Access
14.	Tools complexity
15.	Visibility

hind an attack. Features like motivation, first seen, operations performed, sponsor by, origin country, outcome, and attacker skills have been extracted. More behavioral features can be extracted in the future upon the availability of datasets.

Threat actors always improve their methods due to the dynamic nature of the cyber threat ecosystem. Utilizing behavioral analysis, threat intelligence and attribution procedures may be continuously improved to stay up with new threats. The monitoring and analysis of human and system behavior patterns is done using behavioral characteristics. The following key points contribute to the establishment of context-aware attacker profiles.

The world of cyber threats is dynamic and attackers frequently change their tactics. The development of adaptive threat detection models that can adapt to new and emerging threats is made possible by behavioral aspects. These attributes which comprise regular behavior patterns, access routines, and usage standards, help create detailed user profiles. Contextual awareness reduces false positives by assisting security teams in differentiating between legal activity and potential threats. A context-aware approach to threat identification and response is made possible by using behavioral aspects.

2.8 Datasets for IDS Analysis

KDD Cup 99 dataset was created in the fifth international conference on knowledge discovery and data mining [166]. Creating a network intrusion detector, a prediction model that can distinguish between intrusions and attacks. NSL-KDD dataset [166] was developed in Network Security Laboratory KDD. It contains forty-one features. KDDTrain+, KDDTest21+, and KDD Test+ which includes 125,973, 11,850, and 22,544 records. Aegean Wi-Fi Intrusion dataset (AWID) [167] is the most widely used. It is distinguished by character data and an imbalance between attack and regular data.

Yahoo Web scope S5 dataset [166] consists of annotated anomalous points in real and artificial time series. It examines the precision with which different anomaly categories like outliers and change-points may be detected. Numenta Anomaly Benchmark (NAB) dataset [168] is intended to assess algorithms for detecting anomalies in streaming web applications. It includes more than fifty annotated real-world and synthetic time series data files. Kyoto 2006+ dataset [169] is based

on actual network traffic data collected over three years and classified as normal and attack traffic.

UNSW-NB 15 [170] was generated by the Australian Center for Cyber Security (ACCS) to produce a combination of genuine current normal activities and synthetic contemporary attack behaviors. UNSW Canberra Cyber Range Lab gathered Bot-IoT dataset [171] by simulating a network environment.

The traffic comprises both regular and botnets. ISCX IDS 2012 dataset [172] was developed in 2012. The fundamental concept is based on profiles, lower-level network elements as well accurate descriptions of intrusions. CSE-CIC-IDS2018 dataset introduced the concept of profiles. It had gathered 16,000,000 occurrences in ten days.

This is the latest publicly accessible big data intrusion detection dataset and it encompasses a wide spectrum of attack strategies. This dataset is perfect for testing machine learning models and intrusion detection systems since it replicates real-world network traffic and includes labeled data for both b A summary of the datasets is shown in table 2.3.

2.9 Motivation for Research

Following a thorough review of the literature, it was determined that the threat landscape is always evolving. As a result, there is a requirement to identify the attacker using comprehensive features that include context or motives. Recognizing attackers with behavioral characteristics is a highly significant and difficult task. The key problem in this field is locating a good dataset for feature extraction.
A more thorough understanding of cyber-threat actors is the driving force for the inclusion of behavioral elements in this research. Technical features offer useful information, but they frequently lack the context needed to pinpoint the particular characteristics and intentions of attackers. Threat actors intent and strategy can be inferred from behavioral characteristics.

TABLE 2.3: Datasets for IDS Analysis.

Ref #	Data set	Year	No. of Features
[166]	KDD-Cup99	1998	41
[166]	NSL-KDD	1999	41
[167]	AWID dataset	2015	155
[166]	Yahoo Web scope s5	2015	4 Classes
[168]	NAB dataset	2015	58 Data Streams
[169]	Kyoto 2006+	2006	24
[170]	UNSW NB-15 dataset	2015	49
[171]	BoT IoT dataset	2019	46
[172]	ISCX IDS 2012	2012	16
[172]	CSE-CIC-IDS2018	2018	81

Researchers can more precisely differentiate between actors who may have identical characteristics but operate in different regions with different goals by examining behavioral features.

By providing a more thorough profile that increases detection accuracy and enables proactive defense tactics against known threat actors, this richer, contextual information improves the accuracy of attribution.

Analyzing behavioral features for the identification of patterns and trends associated with specific threat factors is a challenging task in this domain. It can assist in better understanding the goals and objectives of the attacker. Identifying threat actors motivations and targets allows organizations to develop more effective defense mechanisms. Although technical indicators like malware signatures are important, threat actors can swiftly change them. Behavioral features provide a more stable and reliable foundation for attribution. Behavioral analysis facilitates the continuous development of threat intelligence in order to stay up with new threats [129].

2.10 Research Aim and Objectives

To date, the literature has attributed cyber-threat actors using characteristics like TTP, tools, and malware [121]. Attribution has only been performed so far in the research using a limited number of features. This might not offer comprehensive information on the attacker's profile. Focusing on the specific feature set, which also includes the attacker's context, motivations and objectives is necessary to make an accurate assessment of the actor.

The attribution process for cyber-threats will be enhanced by the use of detailed features. The attributes employed in previous research have not proven to be reliable in identifying sophisticated attackers of today. These patterns lack detailed information about the attacker profile, leaving out the objectives and driving forces of the attacker. The attacker's nature and the attack surface are always changing, so it's imperative to have a comprehensive feature set that covers not only tools and tactics but also the context, intentions, and goals of an attack.

In this study, the impact of behavioral traits is investigated. These characteristics may be included in addition to technical for a precise evaluation of the attacker. The impact of hybrid features is also examined. The optimal feature will ultimately be chosen for attribution.

2.11 Research Gaps

Attribution is critical for responding to and preventing similar attacks, but the nature of cyberspace provides several obstacles. It is a complicated and growing field, and closing these gaps is critical to improving CTI efficacy. Here are some examples of potential research gaps. The accuracy and reliability of cyber-attack attribution methodologies must be improved. It is imperative to enhance the precision and dependability of cyber-attack attribution procedures since these approaches are fundamental to pinpointing the accountable entities for attack occurrences. The technical data that is analyzed by current attribution methods like IP addresses, virus signatures, and code similarities can be altered or misconstrued, producing

inaccurate results. The attribution procedure is further complicated by the fact that attackers frequently employ sophisticated methods to conceal their identity, such as proxy servers and credentials theft. To improve these approaches, cutting-edge technology like big data analytic, machine learning, and artificial intelligence must be integrated in order to better evaluate the patterns and behaviors linked to assaults. It is necessary to develop and test more robust techniques, taking into account false positives and negatives as well.

2.11.1 No Standard Format of Reports

There is no standard format of reports and they are mostly unstructured, so it is difficult to extract useful information. The absence of common reporting formats is a significant challenge in this domain. The structure, vocabulary, and content presentation frequently vary greatly, which causes discrepancies and makes information extraction and analysis challenging. Because the reports are unstructured, it is difficult for analysts to evaluate data in a methodical way, spot trends, and determine threat actors with precision. The lack of a standard format makes it more difficult for many companies to share and use vital intelligence, which in turn affects the effectiveness and precision of attempts to attribute cyber-threats.

2.11.2 Limitation of Datasets

There is a limitation of datasets in this domain. The primary challenge in the area of cyber-threat attribution is the problem of class imbalance. Data on cyber-threats is limited for less well-known actors or uncommon attack patterns, but they sometimes include a unequal number of records linked to specific well-known threat actors. Since machine learning models are generally inclined toward classes with more samples, this imbalance skews the training process. Therefore, they may be very accurate for the majority class but fail to detect or attribute attacks to underrepresented some actors. The models capacity to generalize across all classes may be hampered by this imbalance, which could result in less-than-ideal

attribution outcomes, especially for new or under-reported threat actors..There is no benchmark datasets in this field. The problem is crucial in this domain since it could lead security teams to ignore or mistakenly attribute attacks from unknown but potentially harmful actors. To build a more accurate and complete attribution model that can identify a wide variety of threat actors and attack behaviors, class imbalance must be addressed using methods like resampling, synthetic data generation, or algorithms that can manage imbalanced data.

The scarcity of high-quality datasets obtained presents a significant barrier for researchers working in this field. Accessible datasets are frequently imbalanced. Analytical model results may be skewed by this imbalance, making it challenging to reach reliable conclusions. To make matters more complicated, researchers in this discipline are unable to compare the efficacy and accuracy of various attribution approaches because there is no common reference dataset. Previous research work was done on different datasets, so it is difficult to compare various techniques with each other.

2.11.3 Extraction of Features

It is challenging to extract all features from a report as there can be missing values. False flags are commonly used by attackers to confuse investigators, making attribution difficult. It is the need of time to investigate approaches for detecting and mitigating false flag activities in cyber threats, to enhance attribution reliability. Traditional approaches emphasize technological evidence, while behavioral studies of attackers should be prioritized. Investigate strategies for attributing threats based on an examination of attackers' behavioral patterns.
The lack of standardized frameworks for attribution impedes interoperability and consistency across the CTI ecosystem. Propose and assess standardized frameworks for attribution of cyber threats in order to promote a more unified approach among the CTI community. It is critical to create more rigorous and standardized attribution mechanisms. Research could concentrate on improving existing models

or developing new ones that take into account various assault vectors, strategies, and techniques.

2.12 Problem Statement

Cyber-attack attribution is complex due to several factors. Cyber attackers frequently utilize sophisticated strategies, tools, and procedures to mask their identities, making it difficult to correctly identify their origin. Attackers frequently employ false flags or other methods to deceive investigators, resulting in misleading trails and hampering proper attribution. In some situations, the evidence gathered may not be sufficient or precise enough to link an assault to a specific source. Advancements in technology, collaboration between international intelligence and law enforcement agencies, sharing threat intelligence, developing stronger forensic analysis capabilities, and enhancing cyber-security measures all contribute to efforts to improve cyber-attack attribution.

In the past, cyber-attack attribution was done by identifying IoCs found during a forensic investigation. Examples include malware hashes, virus signatures, domain names, and IP addresses. IOCs would then be linked to the TTPs of known threat actors. Attackers impersonate other malicious actors to deflect blame or deliberately carry out false flag operations to harm a rival competitor. These characteristics may not be sufficient to identify the threat actor. Because these attributes can be faked, altered, or shared among various threat actors, it may result in an incomplete or wrong attribution.

Using only technical indications may lead to false positives or false negatives. False positives can falsely accuse innocent persons or misattribute attacks, while false negatives can cause legitimate threats to be dismissed. Technical characteristics do not reveal the motivation or intent behind an attack. Understanding the bigger context, geopolitical considerations, historical behavior, and threat actor aim is required for attribution. By combining these elements a more precise attribution can be obtained.

It is not an easy task to extract attack patterns from cyber-threat intelligence reports. To date, the literature has identified CTA using features like malware, tools, and TTP. Based on context, these characteristics are not able to reliably identify sophisticated attackers of today. These patterns lack detailed information about the attacker profile, leaving out the objectives and driving forces of the attacker.It is essential to have a rich feature set that encompasses not only tools and techniques but also the objectives, intents, and goals of an attack. The impact of behavioral and hybrid features needs to be investigated since the attack surface is constantly shifting. Behavioral characteristics must be included in addition to technical ones for an accurate evaluation of the attacker.

2.13 Research Challenges

Various research issues in this domain must be solved. One of the most significant issues is the massive amount of data. It needs to be collected, processed, and analyzed to generate actionable insight. Another problem is that cyber threats are always developing, necessitating regular updates and agility in information-gathering techniques. Another important research challenge is providing reliable CTI and partnering with other organizations to improve security measures. Overall, the value of CTI is derived from its capacity to offer organizations useful information about prospective threats.

Cyber-attack attribution allows organizations and countries to take appropriate legal or diplomatic action against those involved in the attack. Furthermore, attribution serves as a deterrence for future cyber-attacks by highlighting malicious cyber-activity. To summarize, organizations must use CTI and participate in threat information sharing to improve their security posture and effectively defend against emerging threats. Finally, organizations must recognize its value and invest in the resources and capabilities required to provide accurate and actionable information.

2.14 Research Questions (RQ)

RQ-1 What is the impact of adding detailed technical features in the cyber-attack attribution process?

Objective: The goal of this research question is to examine the detailed technical aspects of the cyber-attack attribution process. Comprehensive, accurate, and trustworthy attack attribution will be aided by this research work.

Methodology: Technical features from unstructured CTI reports will be extracted. So far characteristics like TTP, tools, and malware have been utilized in the research. A novel embedding model called "attack2vec" has been trained on domain-specific embedding to extract features. The additional features have been extracted which are target organization, country, and application to improve the accuracy of CTA detection. These features will be verified using benchmark framework like the MITRE. Attack attribution will be carried out following feature validation to identify the CTA.

RQ-2. What is the impact of behavioral features in cyber-attack attribution?

Objective: Behavioral characteristics are critical in CTA attribution. It includes numerous aspects of the attacker's behavior. This study's goal is to examine the role of behavioral characteristics. The study of extracting these features has not yet been done in research so far. In the process of attributing cyber threats, behavioral features need to be analyzed. It will reveal the context, motivations, and objectives of an assault.

Methodology: The threat-actor encyclopedia dataset, which contains behavioral features will be used to provide an answer to this research question. The goal is to create a CTA profile based on context, motivations, and objectives. There will be a comparison with baseline techniques. In the end, classification algorithms will be used to identify the perpetrator of an attack.

RQ-3 What is the impact of hybrid features (technical and behavioral) in the process of cyber-attack attribution?

Objective: Analyzing the impact of hybrid features is the goal of this research question. They have not yet been incorporated by researchers so far. They will

be analyzed to determine how they affect the attribution process.

Methodology: The extraction of features will make use of the innovative embedding model attack2vec. Different machine/deep learning techniques will be used to determine the performance metrics.

RQ-4 Which set of features are optimal in cyber-attack attribution process?

Objective: The objective of this research question is to identify the optimal feature set for the attack attribution process.

Methodology: The best feature set will provide the answer to this research question. PCA and genetic algorithms among selection techniques will be used. To determine the ideal feature set, machine, and deep learning algorithms will be used.

RQ-5 What is the impact of feature selection techniques in better attack detection in IDS?

Objective: This research question aims to analyze feature selection techniques for improved and precise attack detection for IDS.

Methodology: The methodology uses feature selection techniques to choose the best features for this domain from a variety of datasets for IDS. The classification algorithms determine whether the attack is normal or attack.

2.15 Research Methodology

The research methodology consists of the following phases:

Technical Feature Attribution

Technical features are extracted in the first phase. Data collection, feature extraction, and cyber-attack attribution make up the step-by-step methodology for the extraction of technical features.

Behavioral Feature Attribution

The second phase involves extracting behavioral features. Data gathering, feature analysis, and threat attribution are the phases used for extracting behavioral feature attributes.

Hybrid Feature Attribution

The third phase is the evaluation of hybrid features. To understand the impact, technical and behavioral features are extracted. CTA is categorized using machine/deep learning models based on the attack patterns that are taken from unstructured CTI reports and threat actor encyclopedia dataset.

Optimal Feature Selection

The fourth phase involves selecting the best features. Finding the best features for the process of CTA attribution is the goal of this phase.

Attack Detection in IDS

It involves feature selection techniques in IDS to precisely and accurately detect attacks.

2.16 Research Advantages

Finding the perpetrator of a cyber-attack is a difficult task. Knowing who is responsible for an attack is a crucial step because it enables a nation or business to take precautions against possible future attacks. Finding the attacker from the attack pattern is a challenging undertaking. The goal of this research investigation is to identify the attacker based on its behavioral characteristics. The organizations will be able to accurately identify the attacker with this analysis. The feature set will be more comprehensive as a result of the incorporation of behavioral traits, assisting enterprises in thwarting future attacks. The inclusion of hybrid features will determine the attacker's goals and objectives. Finding the optimal set will assist researchers and businesses. Security companies will be able to attribute cyber-threat actors with the aid of this research investigation. If a feature set is provided, it will attribute cyber-threat actors without a report.

This research work is not primarily focused on a specific sector or industry, it is applicable across various domains, without being limited to any specific one. This study is to offer useful insights and technologies that can be leveraged by organizations in varied fields including manufacturing, defense, critical infrastructure, banking, healthcare, and so on. This adaptability guarantees that the methods

and conclusions derived from our research will be implemented to improve cyber-attack attribution in almost any sector, rendering contribution globally applicable and significant.

Customers who use CTI services and feed as well as security vendors can both profit from this research. It will help security companies identify cyber-attackers based on their attack patterns, i.e., the equipment and methods the attackers used. Organizations will be able to better understand the types of attackers who are interested in breaking into their systems. Our model will be given any unstructured, unseen report and it will identify the attacker. Our work will benefit the defense industry. Forensic investigations conducted by these organizations do not rely on external security experts. They can use this methodology to more precisely attribute cyber threat actors.

2.17 Research Contributions

In this research work, the problem of cyber-attack attribution and the role of IDS for better attack detection has been addressed. Following are the contributions of this research work.

1. A framework for CTA attribution is suggested. A thorough analysis of the literature is done to analyze the methods employed. Important components of the literature review in this field are also emphasized.

2. Moreover, the addition of comprehensive technical features is a research contribution in this domain. It includes a novel concept of adding attributes target country, industry, and application to profile CTA. Earlier studies only used features like tools, malware, and TTP. Up until now, research has not utilized characteristics like target country, organization, and application in this domain.

3. The development of the novel embedding model "attack2vec" is one of this work's main contribution, as general models perform poorly in fields like cyber-security. Datasets from the field of CTI were used to train this model.

4. The impact of analyzing behavioral attributes in this domain is another important contribution. Adding these characteristics is a novel concept in this domain.

To our knowledge, behavioral characteristics have not been used to identify cyber-threat actors.

5. This research investigation analyzes the influence of hybrid features in the cyber-attack attribution process.

6. The selection of the best features is a major contribution in this domain.

7. The creation of a customized dataset that includes hybrid feature will help for future research in this domain.

8. Using CTI feeds for accurate attack detection in IDS is a contribution of this research work.

Chapter 3

Literature Review

3.1 Introduction

In this chapter detailed research literature review is conducted for the cyber-attack attribution process. Also, a detailed study of accurate detection in IDS is reviewed. The role of machine/deep learning is also analyzed. Various techniques proposed by researchers are elaborated comprehensively.

3.2 Cyber-Attack Attribution Literature Review

This paper [119] elaborates on the benefits of domain-specific embedding in the realm of cyber-security. According to the authors, adopting domain-specific embedding can result in high performance. One model is trained on 20,000 unstructured cyber threat intelligence reports, while the second is trained on online pages crawled from Wikipedia. The results demonstrated that a model trained on domain-specific embedding generates better outcomes than web pages crawled from Wikipedia.

A model for attribution of cyber-threat actors was proposed in this research paper [120]. To extract features from unstructured CTI reports, a novel technique called similarity-based vector representation (SIMVER) is proposed. Word2vec

and smoothed binary vector algorithms (SMOBI) are used to compare performance. 238 CTI reports are used for analysis. To attribute various cyber-threat actors, deep learning models are deployed.

It was suggested in this study [121] that cyber-threat actors employ a variety of tools and techniques when targeting enterprises. Changing an attacker's tools can be challenging. As a result, it is crucial to recognize an attacker from their attack patterns. This will assist groups in defending themselves from upcoming assaults. Particularly in the finance sector, identification of these attack patterns is particularly beneficial. In this study, attributes used to identify cyber-threat actors are derived from unstructured CTI information.

In this study, it was hypothesized [154] that most of the information about TTP is available in a human-readable format. It is considered an important feature. Organizations can safeguard themselves against future assaults by extracting it from unstructured data. It forces the attacker to change its tools and techniques, as this is quite a difficult task for the attacker.

In this research work, it was proposed [133] that threat actions can be extracted from threat-related articles. Latent semantic indexing (LSI) and cosine similarity are used to extract features from the dataset. The taxonomy used in this work is MITRE ATT&CK.

In this work [122], different classification approaches to extract features from unstructured text are evaluated. MITRE ATT&CK is used as a benchmark. A tool named reports classification by adversarial tactics and techniques (Rcatt) is developed to extract features from unstructured data. This tool generates reports in STIX format.

There is a lot of raw information present for the CTI [150], extracting it and convert into intelligence can be very useful. To extract it from raw data and draw patterns, the ARM technique is used. It produces rules to find various TTP patterns.

Anti-malware systems are used to detect malicious code or activity within the system [134]. It is out of scope for these systems to detect the attacker behind an

attack and its intent. A lot of raw data for CTI exists in the world today. Manual extracting of information from this raw data is nearly impossible. There is a need to design an automated mechanism to extract useful information from this data and convert it into intelligence.

The studies [123, 124, 135], elaborated that the need for information security is increasing with the development of new technologies and infrastructures. Security analysts and experts face a huge challenge because of increasing security threats. This has led to the emergence of a new field called CTI. This field is gaining popularity in the world today and its importance is growing.

According to [155], modern attackers require expertise to perform cyber-attacks effectively. They employ a variety of preventive measures to avoid detection for an extended length of time. Organizations are concerned about protecting their assets. These attacks have the potential to harm their reputation and cause information leaks. As a result, cyber attribution analysis is a critical and complex process that necessitates a high level of skill.

In this research work [125], it is elaborated that there is no fully automatic and online tool available that extracts meaningful and structured information from raw text. In this work "STIXGEN", an online tool for the development of structured information in STIX format is proposed. This tool will be helpful for organizations to produce structured information.

The work [163] describes levels of attribution. The first level is the host which has started the attack process. The second level is the agent host which has assisted in conducting the attack. The third level is the service provider through which traffic passes. The fourth level is the specific person conducting the attack. The fifth level is the organizations and government agencies who have helped in conducting the attack. The sixth level is from where the assault originated. Attributing the attacker is a powerful preventive defense against the cyber-attacks.

This study [136], proposed an automatic extraction method named "TTPDrill" which automatically extracts threat-related data from unstructured CTI reports. It develops feeds in an STIX format. This method customizes some of the NLP techniques and develops a technique that can automatically extract threat-related

data.

This study [137] proposed a novel approach for analyzing CTI reports and extracting threat-related information from the security corpus. The first contribution is the extraction of features from the CTI reports. This study annotates different threat-related texts. A major contribution is the generation of a 498,000 tag dataset.

In this research work [138], high-level IoC are extracted from threat-related reports. In this work bias correction methods are used to remove biases of data collected from different sources. This method is compared with TTPDrill and it outperforms by producing an Accuracy of 78%.

In the modern world [173], there is a vast amount of text data available. It is difficult to extract useful information. The data is so scarce, that procedures for feature extraction and selection are used as they make the data easier to manage. This study reviews several feature extraction methods and machine learning algorithms for classifying texts.

In this work [139], a framework is proposed known as IL-CyTIS based on the standard STIX format. This work aims to extract the threat actions from CTI reports in a more effective way to attribute the CTA.

In the proposed study [140], different threat actions e.g., tools, file types, and organizations are extracted from the unstructured CTI reports. Reinforcement algorithms are used to evaluate performance measures.

These studies [141, 151], proposed a method for identifying malware and extracting threat actions from CTI reports, honeypot, GitHub, GUN open-source project foundation, and Windows system files. Various machine and deep learning algorithms are used for the classification.

In this investigation [142] a model DeLP was proposed. The goal is to make extraction easier and more accurate to attribute cyber threats. This methodology has made attribution more effective.

An automated mechanism for extracting CTA is proposed in this study paper [174]. Cosine similarity is applied to validate the extracted feature from the

ATT&CK framework. Twenty-seven distinct organizations provide advanced persistent threat (APT) reports.

In this study [175], a threat model is proposed for extracting features from cyber warfare events such as surveillance, data theft, and espionage. This model helps in the extraction of threat actions.

This paper [126], proposed a method for extraction of low-level IoC to attribute CTA. Several frameworks in the literature such as Chain Smith, IOCMiner, and STIXGEN have been proposed for the extraction of threat actions.

In this study [127], a model is proposed for the automatic extraction of threat actions. From APT reports, this model extracts threat actions. TTP is extracted from 521 APT reports. It yields Precision, Recall, and F1-score of 96%, 97%, and 96%, respectively.

In these studies [156, 176], attribution of CTA is proposed. CTI modeling is done using a framework. Identifying various threat categories is the main goal of this work.

For modeling of CTI an automated framework "DLTIF" [157] is developed which can identify various threat types. The aim is to formulate CTI modeling and to identify different threat types.

An innovative method for APT attribution is provided in this paper [143]. The method combines code and the string feature. It uses a bag of word models to represent vectors. Identification of network assaults and threat information is aided by this methodology.

In this work [144], proposed a threat-modeling technique known as "HinCTI". It extracts high-level IOC from CTI data and draws a semantic relationship. It helps in the identification of threat types more accurately and precisely. Comparison with baseline methods is also performed to show the performance of this novel model.

In this approach [177], the Azure Hacker Asset portal is presented to gather CTI data. Various cyber-security platforms offer situational awareness about different parameters. A lot of useful information is present in the dark web. This approach analyzes reports on the dark web to collect insight into CTI for more efficient

utilization.

In the proposed method [128], a honeypot is deployed on an Amazon web service to collect data. After text pre-processing different machine learning algorithms are used to attribute CTA. Out of the used model, support vector machine (SVM) produces a high Accuracy of 94.7%.

In this research work [178], analysis for various types of data is conducted to protect organizations from cyber-attacks. It is important to analyze various types of CTI data. It is now essential to formulate contextual semantic relations in cyber threat text. In this approach, a model known as security open source intelligence framework (OSIF) is developed to analyze CTI unstructured data. Common vulnerabilities and exposures (CVE) dataset is used for cyber-actor profiling.

In this approach [145], a model known as "TIMiner" is proposed for sharing CTI data gathered from social media. Convolutional neural network (CNN) is used to classify various types of IOC from the dataset. This model generates CTI with domain tags.

In the proposed technique [152], the problem of CTA attribution is discussed. It is a challenging task to attribute cyber-threat actors. Attackers mostly conduct attacks behind proxies so it becomes difficult to identify the initiator of attack.

In this study [179], a literature review is conducted on the techniques that extract useful information from unstructured text. A total of 28,484 articles are collected. From the analysis, it is identified that the most useful keywords in the field of NLP are topic classification, keyword identification, and semantic relationship.

In this proposed method [146], a system known as feature-smith is developed to extract features for Android malware. This system improves the overall accuracy of extracting security actions. APT has become a major threat to countries and organizations in the recent past. As it is somewhat difficult for organizations to detect such types of attacks.

A triangle model is developed in this study [129]. It creates a link for attributing CTAs using three criteria (TTP, sector, and tools). The MITRE ATT&CK benchmark framework is used to draw relationships. The suggested paradigm will aid in more precise attribution of CTA.

A methodology [158] for conducting correlation analysis of cyber incidents is described in this article. During a cyber-attack, this framework aids in the correlation of cyber incident occurrences.

Deep learning neural networks are employed in this suggested framework [130] to analyze nation-state attackers. Attributing key players is a difficult undertaking, but it will be useful in the attribution process. These kinds of attacks are often slow-moving and hard to foresee. They may trigger at a specific time, therefore it is difficult to identify them.

In this study[147], several machine learning techniques were used to extract threat actions from various hacker forums. For analysis, only posts written in English were considered. Manual extraction of information from hacker forums is a difficult undertaking due to the high volume of information.

The procedure for attribution is discussed in this article [180]. Various methods exist that aid in proper attribution. The legal and technological components of the process need to be addressed more effectively. It is a challenging task to identify an attacker from an attack pattern in the present world due to the complex attacker nature. The suggested study explains various techniques for cyber-attack attribution.

This research paper[159] established a methodology for analyzing CTI data and producing association graphs. This approach helps organizations to assess potential dangers in the future. Here, a tool is developed for the visualization, analysis, and relationship-building of cyber-threat actions by identifying the organization and attacker responsible for an attack.

Several cyber-attack attribution elements [160] can help with future advancements. It is vital to understand the distinctions between different types, such as strategic and public. In this procedure, the victim's role is critical. Information can be gathered via real-time data or logs.

In proposed studies [131, 148, 149, 153, 181], features are extracted from various sources e.g., dark web and unstructured text. Knowledge base graphs are produced which help in the automated detection of attacks. It is a viable and effective method for converting massive amounts of CTI data into high quality for

analysis.

The effectiveness of high-level assault patterns over low-level ones is examined in this research work [132] to attribute cyber attacks to their perpetrators. The relevant gold standard datasets are necessary to empirically analyze and compare the effectiveness.

This research work [182], presented a thorough framework for CTI implementation. A detailed literature assessment highlighted critical components required for practical CTI, such as data collection, processing, analysis, and dissemination. It could be useful for organizations looking to strengthen their CTI capabilities.

In this paper [161], a novel CTI analysis system, CTI view is developed to automatically extract and analyze the text information of CTI released by security vendors. To be more explicit, this study provides an information extraction strategy based on multiple NLP technologies for extracting APT CTI capabilities.

In this paper [183], a classification strategy for organizing and categorizing existing research works based on the goals of CTI knowledge acquisition is discussed. Current works, including cyber-security-related entities and events, cyber attack TTP, profiles of hackers, and threat hunting methods are elaborated.

In this survey paper [184] techniques for APT attribution are discussed. According to this study, they are divided into four models hierarchical, diamond, Q, and commercial.

The results of this work [185] show that a lightweight technique that leverages fuzzy hashes as natural language input for machine learning classifiers can serve as a credible and fast engine for automated attribution of state-sponsored malware samples for assault analysis .

In this work [162] a framework is developed that asks simple questions at different levels and then combines these primitives to perform the complicated issue of APT attribution. This paradigm aids in reasoning about the process. Furthermore, this design improves the separation of roles, processes, and timing cycles among the various actors participating in the attribution process.

This research [186] presents a mechanism for visually analyzing CTI data using machine learning techniques. The method presented here allows security analysts

to extract relevant patterns from CTI and conduct analysis from numerous angles. The security analyst can recognize common TTP, domains, IP addresses, and file types from previous cyber-attacks. The period of a certain incident is also reported in the visual analysis.

In this paper [187] current state of CTI-based taxonomies and knowledge graphs are investigated. The author has revealed that in recent years, the internet of things (IoT) and cyber-physical systems (CPS) have seen extraordinary growth and numerous success stories.

This study [188] suggests an attack intelligence architecture. An attribution module is proposed that makes use of a variety of deep and machine-learning methods to identify attacks.

This paper [189] presents a solution to facilitate technical attack attribution, implemented as a machine learning model extending the Open-CTI platform. Translated the technical attack attribution problem to the supervised machine learning domain.

In this work [190], a mechanism for identifying CTA by extracting attributes from CTI reports is proposed. Furthermore, an approach for extracting information from unstructured CTI data using natural language processing (NLP) techniques and then identifying CTA using machine learning algorithms is proposed. Using the unique embedding model "Attack2vec" that is trained on domain-specific embeddings, features such as tactics, techniques, tools, malware, target organization/country, and application are extracted. In Table 3.1 comparison of various techniques is shown.

3.2.1 Critical Analysis

A significant issue for attributing CTA is the lack of availability of CTI reports. As the forensic investigation is carried out by security vendors, making these findings publicly available is a significant barrier to maintaining user privacy. Due to dataset limitations, analysis is conducted on unbalanced datasets which may have an impact on accuracy and performance. Different benchmark frameworks are

TABLE 3.1: Comparison of Various Techniques.

Ref.	Features	NLP Tech.	ML Algo.	Dataset (Reports)	Results (Precision)
[129]	TTP,tools, sector	-	-	-	-
[175]	Threat actions	-	-	-	-
[120]	TTP	SIMVER	Neural Network	238	95%
[121]	TTP, tools	LSI, Cosine Similarity	Navies Bayes KNN, DT,RF	327	92%
[122]	Tactics, techniques	LSI, TFIDF word2vec	DT,RF Ada boost	MITRE	79%
[136]	TTP	LSI	BR-Naive Bayes BR-SVM LP-SVM	Threat articles	59.50%
[119]	TTP	SMOBI	XG BOOST	249 & 20,630	55%

utilized for feature validation. For producing accurate results, a single benchmark may be employed.

A report contains a lot of irrelevant information and just a small number of sentences contain information regarding attack patterns, making it difficult to extract meaningful information from this mass of data. The availability of trustworthy reports presents another difficulty. Results may be erroneous and prejudiced if reports are skewed or from unreliable sources.

Since there is no standard format for these reports, this presents another difficulty. Security vendors published it in various formats according to needs and requirements. Therefore, it becomes difficult to retrieve useful information from different

formats. Data is also available in bulk (unstructured reports, blogs, threat warnings, hacker forums, social media, CVE, national vulnerability database (NVD), dark web), so it is difficult to extract meaningful information.
Designing a completely automated mechanism that can depict the complete path of an assault is a problem in the attribution of an actor. Designing a comprehensive automated system for cyber-attack attribution is difficult since attackers use a variety of phases and strategies to carry out attacks. To attribute a threat actor, semi-automated approaches for characterizing an attack flow are available. It is difficult to determine the connections between various incidents of compromise, including TTP malware and tools, in this field.

Evaluating the outcomes of numerous research initiatives that have been conducted on various datasets is a very difficult task. There are only a certain number of datasets available in this business. As a result, some important details will probably be missed. There have only been a few features employed in the research so far. They do not offer in-depth information on the attacker's profile. However, little has been done to create a context-aware profile of the attacker up until this point. It is necessary as technology advances, the attack surface, and attacker context are changing very rapidly.

It is also vital to assess the objectives and drives of the attacker. These factors may aid in this process as they give the attacker more information, motivations, and goals. The process of attribution for CTA is complicated. The datasets to extract behavioral traits is still not available. For information on cyber threats, there is a ton of raw data that may be acquired from various sources. It is challenging to draw insight from such a large volume of data.

So far a small number of features (TTP, tools) have been extracted for CTA attribution. There are other important attributes such as target organization/ country/application which may improve the cyber-attack attribution process. This may provide detailed information about the attacker's profile. This study [119] proposed an embedding model known as SMOBI. This technique [121] extracted tools and TTP from unstructured reports. LSI is modified according to the author but is not explained in detail. The authors [120] extracted tools and TTP from

unstructured reports. They proposed an embedding model known as SIMVER. Detailed feature set has not been used in this work. Various research generates results on different datasets, so they cannot be compared.

So far, no fully automated mechanism has been designed for cyber-attack attribution. There are semi-automated mechanisms that design the attack flow process. Analysis has been conducted on different frameworks such as the diamond model, CKC, F2T2EA, and MITRE framework.

There is a need for a single benchmark, based on which experimentation may be performed for comparative analysis. In the literature, so far relationship between different TTP is drawn. It will be better to provide a more detailed relationship between various attributes. Pros and cons of various techniques are illustrated in table 3.2.

3.2.2 Important Aspects of Literature Review

Important aspects in this domain like NLP and machine learning techniques, features, results generated, performance metrics, cyber-threat actors, tools, and frameworks developed in this domain are highlighted below. These aspects are as follows.

Q-1. Which Natural Language Processing (NLP) techniques have been effective in this domain?

(NLP) is a branch of artificial intelligence (AI) that studies how computers and human language interact. It entails creating models and algorithms that allow machines to meaningfully and practically comprehend, interpret, produce, and work with human language.

NLP is used for text cleaning processes like removal of stop words, punctuation, tokenization, stemming, lemmatization, and extraction of features. Effective techniques used in this domain are frequency-based, i.e., term frequency - inverse

TABLE 3.2: Pros and Cons of Techniques.

Ref	Pros	Cons
[129]	Creative methodology. Practical to implement. Robust framework. Detailed work flow.	Limited scope. Dependent on accurate data. Dependent on static indicators.
[175]	Comprehensive approach. Real time examples.	Heavily focused on state actors. Privacy implications.
[120]	High level of information. Innovative approach. Domain specific embeddings. Feature engineering.	Limited dataset. Overfitting. Relies on quality of data. Computational complexity. Pre-processing challenges.
[121]	Detailed analysis of techniques for imbalanced data. Best metrics like ROC used for imbalanced datasets. Addresses challenges in mutli-label classification.	No case study to prove results. Limited discussion on hybrid methods. Not adressing on computational complexity.
[122]	Integrates multiple sources of data. More accurate identification of threat actors. Broad applicability to various industries. Rich dataset.	Increased complexity. Data management challenges. Issues of data quality. Scalability issues.
[136]	Improved cyber-threat attribution. Comprehensive dataset. Industry wise applicability. Detailed methodology.	Complexity in implementation. Data availability issues. Computational requirements. Potential biases.
[119]	Innovative approach. Effective use of NLP and ML.	Dataset limitation Manual label required. Dependence on report quality. Lack of real time analysis.

document frequency (TF-IDF), and context-based, i.e., LSI etc. Some novel models have also been developed by the researchers for extraction of features.

After the literature survey, it is identified that TF-IDF, latent semantic indexing (LSI), and named Frequency of NLP techniques are shown in figure 3.1.

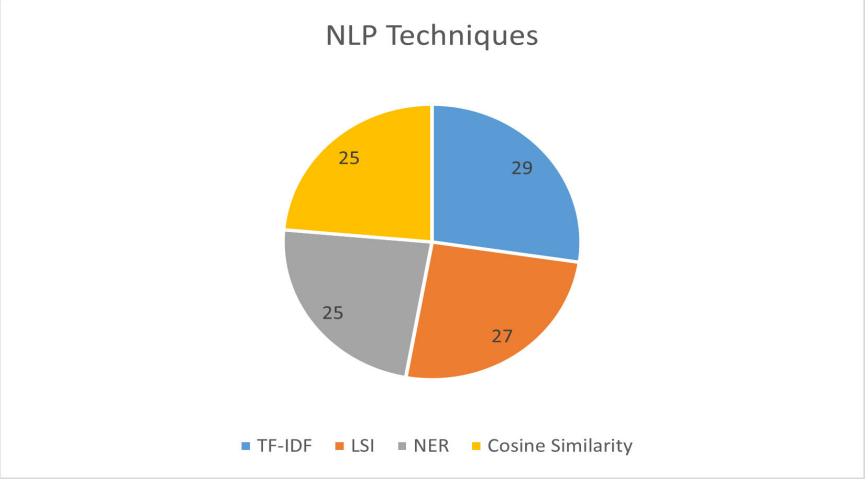


FIGURE 3.1: NLP Techniques used in the Literature.

Q-2. Which machine/deep learning models have been effective in this domain?

In this research question, machine/deep learning models used in this domain have been highlighted. The effective techniques used in the literature are random forest, deep learning neural networks, decision trees, long short term memory (LSTM), and SVM models. They are effective in producing high results. To solve this problem in the literature various techniques like BR-SVM, BR-DT, LP-SVM, and LP-Naive Bayes have been applied. The frequency of techniques used in the literature is shown in figure 3.2.

Q-3. What kind of performance metrics have been used in the literature & which metric is most used?

The performance metrics mostly used in this domain are Accuracy, Precision, Recall, F1-measure, confidence, support, and lift. Precision is the most used and effective metric used in the literature. Other used metrics are Recall, F1-measure, and accuracy. Confidence, support, and lift are metrics for finding regularities and patterns in the datasets. The frequency of performance metrics used in different

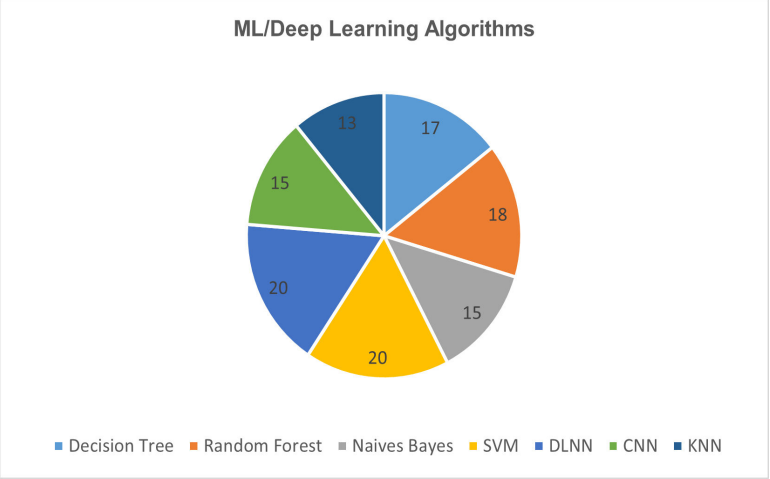


FIGURE 3.2: Frequency of ML/Deep Learning Algorithms.

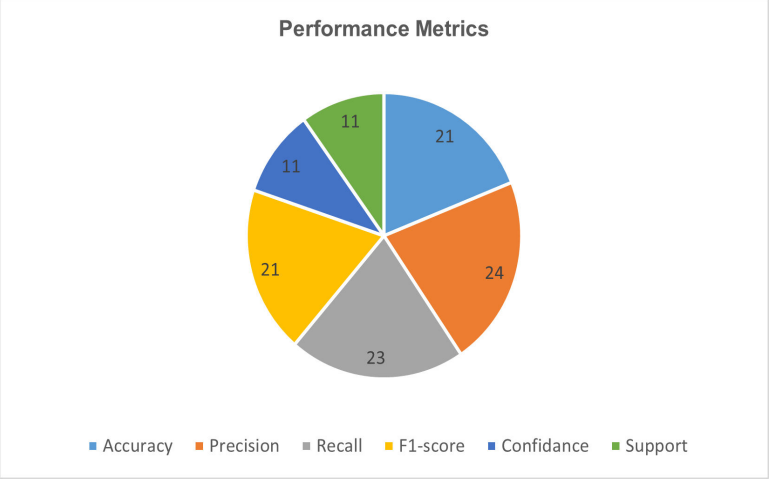


FIGURE 3.3: Frequency of Performance Metrics.

work is shown in figure 3.3.

Q-4. Which features have been used in this domain & are considered most important for cyber-attack attribution?

There are two types of features commonly used in this domain. High-Level and Low-Level IoC. After the literature review, it is evident that TTP is the most used feature in this domain. For experiments, now researchers are mostly focusing on high-level IoC, as their impact is high and everlasting. Identifying them can force the attackers to change their tools which is a very difficult task. The frequency of features used in various works is shown in figure 3.4.

Q-5. What results have been generated by different techniques?

The results are shown in table 3.3.

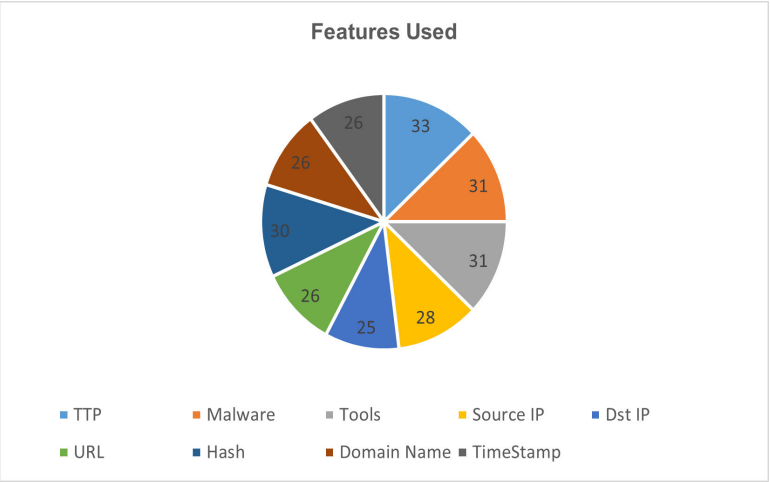


FIGURE 3.4: Frequency of Features.

Q-6. Which feature selection techniques have been used in literature?

TABLE 3.3: Attribution Results.

Author/Ref	Accuracy (%)	Precision(%)	Recall(%)	F1-Measure(%)
S. Naveen[120]	86.5	95.4	83.3	87.9
U. Noor[121]	94	92	89	89
L. Perry[119]	58.4	55	52.4	-

Information gain has been mostly used in this area. It is the commonly used technique identified for feature selection. Its role is to identify the most effective feature from the dataset.

Q-7. What are the most used benchmark frameworks in the literature?

Different benchmarks used have been identified in this research question. MITRE ATT&CK is the most used framework in this domain. The purpose of these benchmark frameworks is to validate a feature. CKC is also used by the researchers before the development of the MITRE framework. CVE database is also used for the extraction of features.

Q-8. Which cyber-threat actors (CTA) are mostly used in the research?

Cyber-threat actors are the attacker or person behind a cyber-attack. In the literature a CTA has been used by different names, so aliases are also identified by

the researchers.0.25cm

Mostly used CTA in the research are APT28, Lazarus, Turla, Oil Rig, APT17, Fin7, APT29, menu Pass, Deep panda, APT1, admin338, Rocket Kitten, APT12, APT16, APT18, APT30, APT 32, APT34, Equation, FIN5, FIN6, Gameredon, Rocket Kitten, CGMAN, Group5, Ke3chang, Lotus Blossom, Magic Hound, Moafee, Winntie, APT3, APT17, APT28, Molerats, Bronze Butler, Carbanak, Cleaver, Dark hotel, Copy Kittens, Dragonfly, Dragon OK, Dust Storm, Fin10, Copy Kittens.

Q-9. What are the most important tools, standards, expressions, and information-sharing platforms used in the literature? 0.25cm

In this research question, the identification tools, expressions, and information-sharing platforms used in the literature have been identified. STIX is considered the most used expression for cyber threat intelligence. Trusted Automated Exchange of Indicator Information (TAXII) and Open-IOC are the platforms for extracting threat feeds. Open-source intelligence (OSNIT) is also used by the researchers. It is an open-source repository for collecting information about the attacker.

Q-10. What are the novel frameworks/tools developed in this domain?

In this research question the novel frameworks and tools developed have been identified. Some of them are explained below.

STIXGEN: - It is a tool developed for the generation of CTI feed in a more detailed and comprehensive manner from raw text data. It will ensure the sharing and availability of CTI feeds among various organizations.

Action-Miner: - The goal of this tool is to extract low-level IoC from CTI feeds more efficiently and accurately as compared to other tools.

ATIS: - Automated threat intelligence fusion framework considers different sources to create intelligence from various data sources. It is a collection tool to collect meaningful information and draw relationships from this data.

Six-gill: - It is a tool used in the dark web that collects hacker information from different sources. This tool aims to extract features from the dark web.

TTP-Drill: - This tool extracts threat action, and then converts it into STIX format from unstructured CTI reports.

IoCMiner: - It is a framework for extracting IoC from unstructured text from Twitter.

Feature Smith: - It is a system to generates a feature set for detecting malware on the Android platform.

SMOBI: - It is an improved bag of word models. It assigns weights to each entry in the model. Then it finds words in the vocabulary with similar embedding based on cosine similarity.

SIMVER: - It is a way of representing neural embedding. Use similar words, if a word is available in the datasets, assign the current index in the matrix. It uses the skip-gram model.

Q-11. Which security vendor and external sources are mostly used by the researchers?

In this question security vendors and external sources used in this domain are identified. It is important to know the reliable vendors and sources used by the researchers. Symantec, fire-eye, crowd strike, and trend-micro are mostly used. For extraction of threat actions from raw data twitter stream is mostly used.

3.3 IDS Literature Review

In this study [191], a classifier approach for NIDS by using a tree algorithm is applied for detecting attacks. The author has proposed a combining tree classifier approach for detecting network attacks. First implemented individual tree algorithms (Random tree, C4.5, NB Tree) on NSL-KDD data to know the accuracy of individual algorithms for detecting attacks. Then different algorithms are combined to determine the accuracy.

In this study [192], the IDS framework is proposed. The NSL-KDD dataset is

used as a benchmark. The wrapper approach was used for feature scaling. After applying this technique, 16 feature sets were used to obtain results instead of the actual 41 features.

In this study [193], ml techniques are used to detect security attacks. SVM is utilized in this strategy to enhance the accuracy of attack detection. The NSL-KDD dataset is employed. The 41-feature set is separated into three categories: basic, content, and traffic.

The study [194] investigated the viability of merging fuzzy logic with machine learning techniques to detect intrusions. The suggested architecture mined fuzzy association rules using machine learning methods, extracting the best possible rules using a genetic algorithm.

The author [195], presented a novel concept for attack detection. The proposed study proved that if k-means clustering is applied, IDS accuracy improves in detecting attacks. This model performs best when given multiple clusters that correspond to the number of data types in the dataset. When the number of clusters changes, the performance of K-means degrades.

In this study [196], it has been elaborated that entropy can detect abnormal network behavior but with a high false rate. The SVM model can classify traffic as normal or malicious by learning different features of the network. The goal of this study is to overcome the shortcomings of entropy and SVM. So, the authors produced a hybrid solution for attack detection. The dataset used in this proposed method is provided by MIT Lincoln Laboratory.

In this research work [197], authors have used k-means with a naive bayes algorithm in IDS. This study shows that the k-means is not appropriate for anomaly detection because in some cases (especially in passive and observatory attacks, etc.) intrusion behavior is almost the same as normal. If the k-means algorithm is used with naive bayes, the detection rate increases with low false alarms. Authors have conducted experiments on the Kyoto 2006+ dataset.

In this study [198], a detailed review of anomaly-based detection in which single, hybrid, and ensemble machine learning models are used to evaluate different data

sets. This comparison shows that both models provide higher accuracy and detection rates.

This study [199], presented a hybrid system that uses two detection systems. KDD-Cup dataset is used for the training of the system and about 30,000 files from window XP are used to perform experiments.

Using the NSL-KDD dataset, these studies [200, 201] compared the performance of two supervised machine learning models. Four ML algorithms are used to create an ensemble model. Two data sets, UNSW NB-15 and UGR-16, random forest, KNN, SVM, and logistic regression are applied on emulated and actual network traffic.

In this study [202], to detect intrusion in a computer network four ML algorithms are applied to the KDD Cup dataset to analyze performance. These algorithms performed best on test datasets.

These papers [203–205] investigate ML/DLNN models for IDS. Various machine learning algorithms are used and their performance is tested using KDD cup data in terms of various performance metrics. Random forest performs well with overall 94% accuracy.

In this paper [206], a one-dimensional CNN-based deep learning method for creating an effective and flexible IDS is presented. Normal and abnormal network traffic are classified and labeled for supervised learning in the 1D-CNN. Tested this proposed model using the UNSW-NB15 IDS dataset to demonstrate the efficacy of the approach.

This study [207] key contribution is the presentation of a HIDS that builds on the well-known consolidated tree construction (CTC) technique to effectively handle class-imbalanced data. At the pre-processing step, a supervised relative random sampling (SRRS) technique was developed to get a balanced sample from a high-class imbalanced dataset. In addition, an advanced multi-class feature reduction approach was devised and built as a filter element to deliver the best standout features from IDS datasets for effective intrusion detection.

This investigation [208] improves IDS detection mechanisms through two processes: a DLNN model with new features for threat detection and a comprehensive

solution that combines the DLNN model and PCA to increase security and performance. According to analytical and software results, the suggested detection system, which integrates DLNN, PCA, statistical, and knowledge-based methodologies, surpasses existing IDS.

In this paper, IMIDS [209] was proposed as an intelligent IDS to protect Internet of Things (IoT) devices. The heart of IMIDS is a lightweight CNN model that can classify a wide range of cyber threats.
This article proposes an attack data generator driven by a conditional generative adversarial network to assist the problem of a shortage of training data. IMIDS beats its competitors in the testing, detecting nine different types of cyber-attacks with an average F1-measure of 97.22%.

This work [210] presented the Intrusion Tree (IntruD-Tree) machine-learning-based model, which first considers the ranking of security elements according to their value. This approach reduces computing complexity by reducing feature dimensions, making it beneficial in terms of prediction accuracy for previously unseen test scenarios. Finally, experiments were run on cyber-security datasets to test the effectiveness of this model and the Precision, Recall, F1-score, and ROC values were calculated. In table 3.4 results of various techniques are compared.

3.3.1 Critical Analysis

There are various research issues in accurate attack detection in IDS employing ML techniques. Some of the most common obstacles that researchers face in this field are as follows.
Many ML models necessitate a significant amount of computational power, which can be a barrier to real-time processing in large-scale network setups. A significant difficulty is developing efficient real-time algorithms. It is a constant challenge to strike a balance between properly detecting anomalies (attacks) and minimizing false positives. A high percentage of false positives might cause alert fatigue and a loss of faith in the IDS. It is critical to create ML models that can scale with the increasing number and complexity of network data.

TABLE 3.4: Comparison of Different Techniques.

Author/Year	Dataset	Technique	Results (%)
A. Alzahrani et al. /2021	NSL-KDD	XGBoost	Precision 92 Recall 89 F1-Measure 90
V. Pai et al. /2021	NSL-KDD	RF	Accuracy 91 Precision 92 Recall 90 F1-Measure 92
A. Halimaa et al. /2019	NSL-KDD	SVM	Accuracy 93
K. Abu et al/2019	CICIDS-2018	ANN	Accuracy 91
M. Fawareh et al. 2022	CICIDS-2018	DLNN-PCA	Accuracy 96
J. Kim et. al. / 2019	CICIDS-2018	CNN	Accuracy 95
V. Kanimozhi et al. /2019	CICIDS-2018	ANN, RF, KNN, SVM, Ada-boost	Accuracy 96 Precision 90 Recall 95 F1- Score 90

The problem is to create models that remain successful as the volume of data grows. Some ML models' lack of interoperability makes it difficult for security analysts to comprehend why a given choice was taken. Building trust in the system requires development of interoperable models. There are resource limits in many network scenarios, particularly in IoT or edge computing. It is difficult to create lightweight ML models that can function efficiently in resource-constrained contexts. It is difficult to ensure that ML models can adapt to changes in network behavior over time. Continuous learning techniques that allow models to update themselves without having to retrain from scratch should be investigated further. Researchers are actively working to address these issues to improve the performance of ML-based intrusion detection systems in dynamic and complex network environments.

Many machine learning models, particularly complicated ones DLNN, lack interoperability, making it difficult to grasp the reasoning behind their judgments. This is a challenge for security analysts, who may find it difficult to comprehend and trust the data.

The quality of training data is critical for machine learning models. Inaccurate models might result if the training dataset is biased or incomplete. Preprocessing issues like coping with imbalanced datasets and noisy data, might have an impact on the performance of ML-based IDS. Adversarial attacks, in which attackers modify input data to mislead the system, can make ML models vulnerable. Adversarial assaults, if not addressed properly, might undermine the effectiveness of ML-based IDS.

In conclusion, while ML-based IDS has shown significant promise in terms of improving the accuracy and adaptability of attack detection, addressing issues (interoperability, data quality, adversarial attacks, and resource requirements) is critical for their widespread and effective deployment in real-world scenarios. Ongoing research efforts are aimed at overcoming these obstacles and boosting the capabilities of ML-based intrusion detection systems.

Imbalanced datasets are a major issue in this domain. When compared to normal behavior, a small number of real attacks can lead to an imbalanced dataset. This has an impact on the ML model's capacity to generalize well to real-world circumstances. New attacks are emerging very rapidly. Cyber-attacks are continually developing, and new attack methods appear regularly. ML models must adjust to these changes and learn from new patterns, making maintaining high accuracy difficult.

Attackers may purposefully modify training data to deceive ML models. The focus of research should be on constructing robust models that are resistant to adversarial attacks. It is critical to identify significant traits for assault detection. It is difficult to choose and extract features that represent the essence of both regular and malicious behavior, especially in complicated and high-dimensional data.

A critical analysis of studies for attack detection in IDS utilizing ML approaches includes assessing the literature's strengths, limitations, and contributions.

ML approaches have shown great Accuracy in identifying numerous sorts of assaults, both known and unknown. They can process enormous amounts of data efficiently, making them suited for real-time detection in dynamic network contexts. ML models can adapt to changing attack patterns, providing a level of flexibility that classic rule-based intrusion detection systems may lack. They learn and update their expertise over time, allowing them to detect emerging risks more effectively. ML is skilled at identifying relevant aspects from raw data, allowing for the detection of minor patterns indicative of assaults. Feature extraction can improve detection Accuracy while decreasing false positives.

Chapter 4

Research Methodology

4.1 Introduction

It is critical to take into account the possible exploitation in a variety of scenarios when discussing cyber attack attribution. Tools are utilized appropriately and do not contribute to unjustified breaches of privacy or misuse of sensitive information would be among the ethical considerations. It is ensured that the use of behavioral data in CTA attribution is both beneficial and respects people's privacy and rights. It is crucial to address ethical issues. The requirement for cyber-security precautions and the defense of core ethical ideals must be balanced. People should be made aware of the gathering of their behavioral data and be allowed to expressively consent to its use in cyber-security measures. To safeguard behavioral data against unauthorized access, breaches, or cyber-attacks that can jeopardize people's privacy. Strict security measures must also be in place.

This dissertation methodology consists of the following phases.

i. Technical feature attribution

ii. Behavioral feature attribution

iii. Hybrid feature attribution

iv. Optimal feature selection

v. Attack detection in IDS

4.2 Technical Feature Attribution

The extraction of technical features consists of three phases, i.e., data collection, feature extraction and threat attribution. To attribute CTA a framework is proposed.

4.2.1 Proposed Framework for Technical Features

Figure 4.1 illustrates the proposed framework. It consists of three stages. Data is collected in the first phase in the form of unstructured CTI reports. Data is stored in the CTI corpus manager. The feature analysis or extraction phase comes next. Text pre-processing like removal of stop words, punctuation, special characters, tokenization, and lemmatization is performed in this phase to clean the text. The Nltk package is used to remove stop words.

The feature extraction engine phase comes next. During this step, features are extracted from text using the novel embedding model attack2vec, which has been trained on domain-specific embedding. The vocabulary size of the model is approximately two million. Then these extracted features are saved in an index archive. Using cosine similarity, these derived features are compared to benchmark framework like MITRE ATT&CK.

Following feature validation, the CTA attribution phase is initiated, during which the CTA is identified using various classification approaches like decision tree, random forest, and support vector machine. The purpose of CTA attribution is to identify the actual attacker who has carried out the attack.

4.2.2 Data Flow for Technical Features

Figure 4.2 depicts a data flow diagram. CTI reports are used as input in this flow. The text is then cleaned using text pre-processing techniques such as the removal

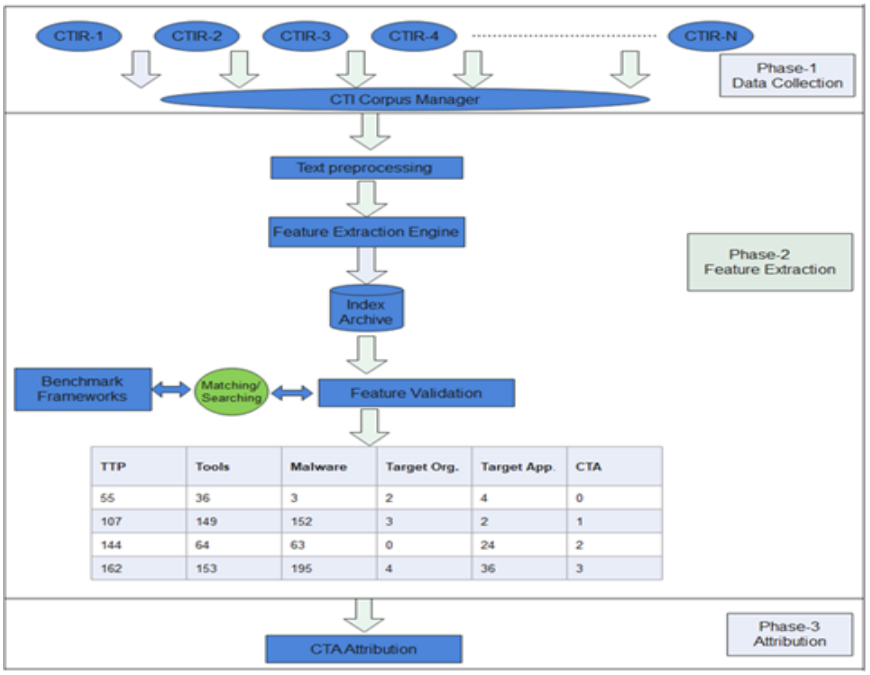


FIGURE 4.1: Proposed Framework (For Technical Features).

of stop words, punctuation, and lemmatization. Then, using the innovative embedding model attack2vec, features are retrieved. Following feature extraction, these are evaluated against benchmark frameworks e.g., MITRE ATT&CK, CAPEC, APT group, and operations using cosine similarity. If a feature matches one of the benchmark frameworks, it is included in the corpus; otherwise, the feature is rejected and the search for the next word begins. The next step after feature validation is CTA attribution. Classification is carried out to identify the perpetrators of incident. Data for the classification algorithm is separated into training and testing. Following that, CTA attribution is carried out.

4.2.3 Phases for Extraction of Technical Features

It consists of following phases.

4.2.3.1 Data Collection

The goal of this phase is to collect unstructured CTI reports from various sources. Data was gathered from studies released by the research community, security firms,

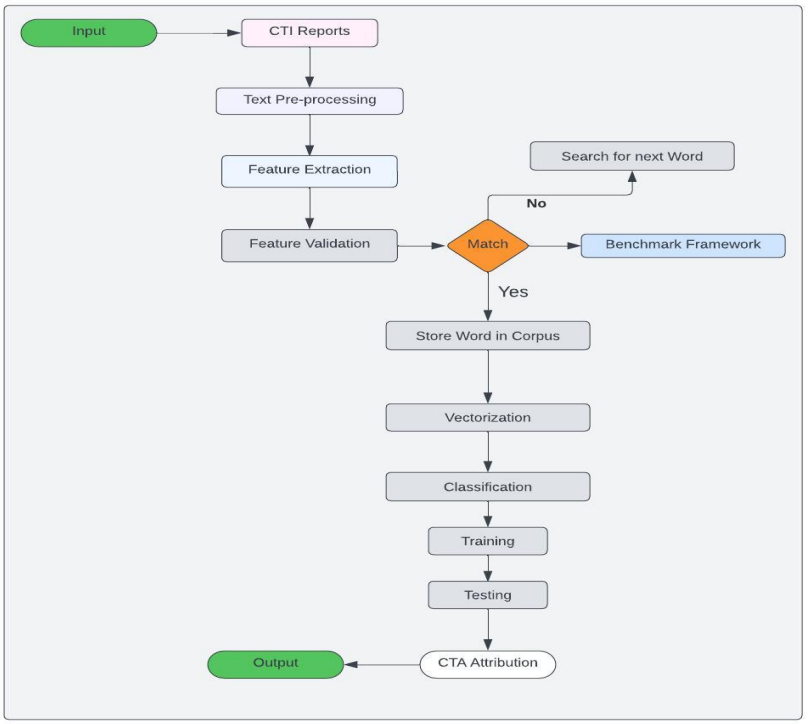


FIGURE 4.2: Data Flow Diagram for technical Features.

and a Google programmable search engine. The number of datasets available in this domain is limited. So, the goal of this phase is to collect datasets to conduct experiments. So, in this phase around 27,000 CTI reports are collected for experimentation having twelve cyber-threat actors.

4.2.3.2 Feature Extraction

Feature extraction refers to the process of discovering and selecting important and meaningful traits or attributes from raw data to aid in the identification and analysis of potential cyber threats. It is a critical phase in the development of models for threat detection, classification, and prediction. The feature extraction procedure is divided into three parts. The first stage is text pre-processing, often known as text cleaning.

The second step is feature extraction which involves training an embedding model known as attack2vec. It is trained on domain-specific embedding. The third step is semantic mapping, which uses cosine similarity to detect similarities between

distinct documents. This stage validates extracted features against benchmark frameworks. The extraction of features is an important step in the process since the quality of features has a direct impact on the effectiveness of threat detection and mitigation.

4.2.3.3 Text Pre-processing

Text pre-processing is an important stage in NLP that entails converting raw text input into a more usable and analyzable format for ML and other NLP activities. Its goal is to clean, standardize, and organize text data to improve its quality and prepare it for future analysis. Processes like converting text to lowercase, removal of stop words, punctuation, special characters, tokenization, and lemmatization have been performed. These words may affect the performance of the model, so it is necessary to remove common words and clean the text.

In order to extract valuable features, this study used a variety of NLP approaches to preprocess and clean unstructured text data from CTI reports. Every stage of the text-cleaning procedure is selected with care to strengthen the models capacity to accurately ascribe CTA based on textual patterns and to improve the quality of data. Preprocessing began with the removal of stop words. Words like "and," "the," "is," and "it," which are often used in the English language, have little semantic significance by themselves but can introduce extraneous noise into text data. We make sure that the analysis concentrates on more important phrases that are pertinent to the circumstances surrounding the threat actor's activities or motivations by removing these words.

Punctuation and special characters, which frequently occur in unstructured text but offer no helpful information for threat attribution, were then eliminated. Because different reports may employ different punctuation styles, this process also standardizes the text data. To maintain consistency, special characters like hashtags, asterisks, and other symbols were removed from the text, which made it simpler to use feature extraction techniques efficiently. After that, the text was changed to lowercase, which is a crucial step in preserving uniformity throughout

the dataset. We avoid unnecessary feature representation by decreasing the case of each phrase, ensuring that capitalization variants (such as "Malware" vs. "malware") are recognized as equivalent terms. In text classification tasks, this step is especially crucial because it minimizes the vocabulary size and stops the model from treating different capitalization as distinct entities.

The next step was tokenization, which included dividing the text into discrete words or tokens. Since each token represents a sentence meaning unit, tokenization serves as the basis for additional text processing. We employed word-level tokenization in this study to record particular phrases related to threat actors, instruments, strategies, and tactics. In order to facilitate the analysis of technical jargon and abbreviations commonly seen in CTI reports, tokenization parameters were specified to ignore non-alphanumeric characters and to guarantee consistency in the breakdown of difficult phrases.

Lastly, each word was reduced to its root or base form via lemmatization. A linguistic technique called lemmatization takes into account each word's context and returns inflected forms (such as "running," "ran," and "runs") to their base form ("run"). Lemmatization is more accurate and context-aware than stemming, which only eliminates suffixes and could produce erroneous word forms. This makes it perfect for extracting consistent and coherent features. Lemmatization enhanced the quality of the dataset in this study by lowering word variations, which helps create a model that more accurately detects patterns.

To customize the cleaning procedure for CTI data, each of these preprocessing processes was carried out with particular parameter settings. When combined, these NLP techniques enhance the text data quality and relevance, making it possible to extract technical and behavioral aspects more precisely. This, in turn, improves the attribution model's ability to recognize and distinguish between CTA.

Each of these processes tries to improve text data quality and consistency, making it more suited for analysis and modeling. Text pre-processing is not always a one-size-fits-all procedure; the stages required might be tailored to the task at hand and the nature of text input. For the removal of stop words NLTK library is used in Python. The process of text pre-processing is shown in figure 4.3.

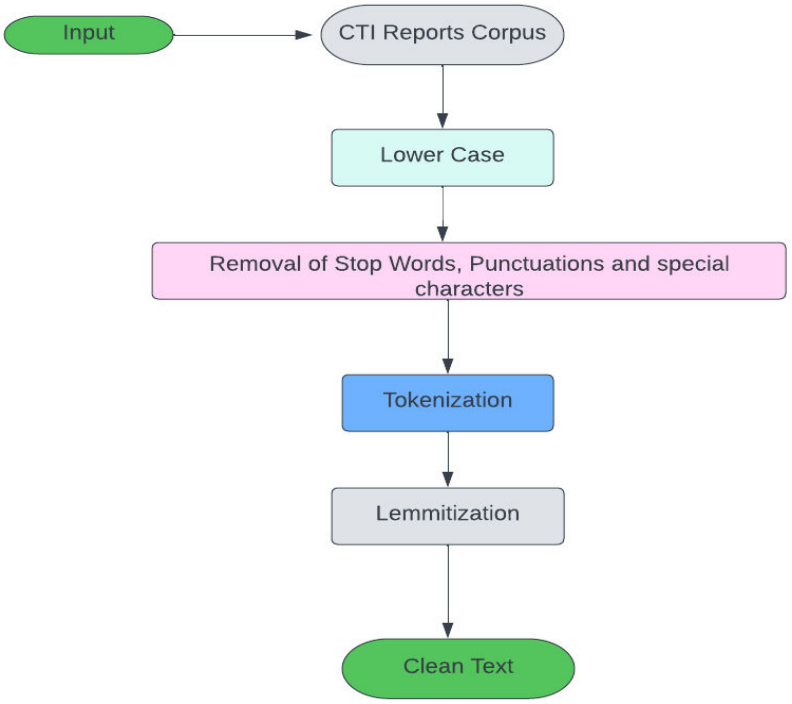


FIGURE 4.3: Text Pre-processing.

4.2.3.4 Attack2vec Embedding Model

Following text cleaning, the next objective is to extract features from unstructured CTI reports. For this, a unique embedding model known as "attack2vec" was created. It is based on the cutting-edge word2vec model. General embedding models do not yield satisfactory results in domain-specific environments such as cyber-security. A model trained on domain-specific embedding is required. Because the word2vec model is trained on Wikipedia pages, it does not perform well in domain-specific domains. Attack2vec is trained on domain-specific embedding to circumvent this constraint.

Datasets from cyber-security fields are collected for training the attack2vec model on domain-specific embedding. The model has a vocabulary of two million words. The datasets used for training is shown below.

a. 20,630 CTI reports [119]

b. 18,257 CTI reports [138]

c. 17,000 CTI reports [128]

e. Malware Samples [142, 143, 146, 185]

The Attack2vec model is made up of three layers: input, hidden, and output. Weights have been assigned. It is illustrated below. The model input to the neural network is displayed in figure 4.4.

Attack2vec Algorithm

Input

F - CTI corpus

WF - Word corpus, set of words in the corpus

WD - Word corpus after text pre-processing

Wu – Unique words vocabulary

Output

V: Vector representation

Initialize WF of size |max | with words from text files F

x:= 0

WD ← NLTK (WF [max])

For i: = 0 to |max| do

temp: = WD [i]

If temp: = ∉ S

S [x] := temp

x: = x +1

i ← i+1

V ← S

The technical features extracted in this phase as shown in table 4.1 are TTP, tools, malware, target country, organization, and application.

4.2.3.5 Semantic Mapping

In this step, extracted features are validated against benchmark frameworks like MITRE ATT&CK, global industry standards, and alternative country and application names from Wikipedia. The extracted feature is tested against framework benchmarks. If it is validated, it is included in the corpus. TTP and malware are validated from the MITRE framework. The feature target industry is compared

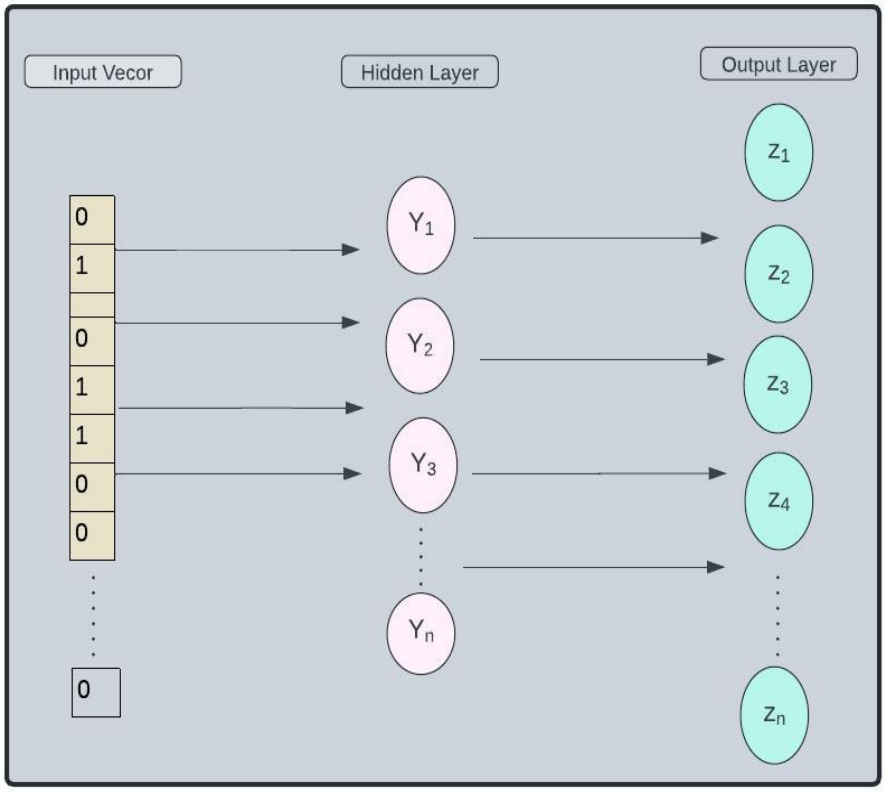


FIGURE 4.4: Attack2vec Neural Network.

to sp-global [211], country names are compared to Wikipedia alternative country names [212], and software is compared to Wikipedia list of software's [213]. Cosine similarity is used to validate the feature set against the corpus. Figure 4.5 depicts the procedure.

A customized semantic mapping for identifying CTA based on unstructured CTI reports is developed in this research work. The mapping for features like TTP, tools and malware exists but features like target country/organization/ application for which mapping does not exists. Existing mappings frequently fail to capture the contextual variances found in data, which are critical for reliable threat attribution.

Furthermore, the continually changing cyber threat landscape demands an adaptive solution, as domain-specific mappings may not be updated frequently. This proprietary mapping improves the accuracy of attribution method, resulting in a more exact alignment with the dataset. This personalized approach not only

TABLE 4.1: Features Extracted.

Sr. #	Features	Remarks
1.	CTA	Class
2.	TTP	High level IoC
3.	Malware	High level IoC
4.	Tools	High level IoC
5.	Target organization	-
6.	Target country	-
7.	Target application	-

addresses the limits of existing mappings, but it also contributes to the field by introducing a new method that can be used in future study with similar unstructured datasets.

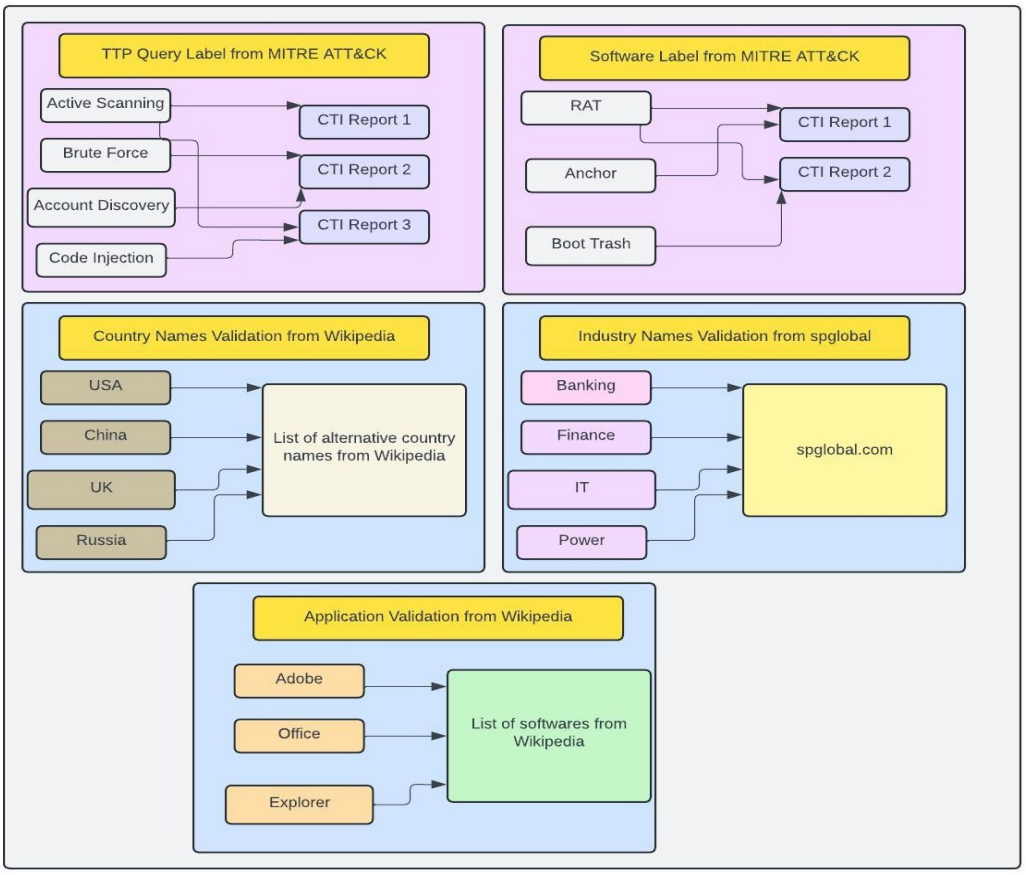


FIGURE 4.5: Semantic Mapping of Various Features.

4.3 Cyber-Threat Attribution

In this step various ML/Deep learning models are used. These are described below.

Decision Tree It is one of the simplest classification algorithms. It is well suited for categorical types of datasets. It is a supervised ml algorithm that is used to solve classification and regression problems. In it a tree is made which has two types of nodes; one is a root node and the other is a leaf node. Prediction starts from the root node. A major challenge in this algorithm is the selection of the root node. Different algorithms such as ID3, CHAID, C4.5, CART, and MAR can be used in it depending on the classification problem. The logic of this algorithm is easy, so it is easy to understand.

It starts from the root node and goes down to the leaf node for the selection of an attribute. Information gain and the Gini index are used for feature selection. It requires less data cleaning, and it helps in identifying all possible outcomes. Entropy and information gain are important factors in deciding the appropriate attribute to split the dataset at each node. The primary functions of entropy and information gain in decision trees are as follows.

Entropy measures impurity or disorder in a dataset. It quantifies randomness in the data class distribution. A dataset with low entropy is pure (all data points belong to the same class), whereas a dataset with high entropy is impure. The formula is defined as:

$$\text{Entropy}(S) = -\sum_{i=1}^{c} p_i \log_2(p_i)$$

Information gain is utilized to determine which attribute to split on at each node of the decision tree. It calculates the amount of information acquired from splitting the dataset depending on a specific attribute. The splitting criterion is set based on which property maximizes information gain when split. The formula is defined

as:

$$\text{Information Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{values}(A)} \frac{|S_v|}{|S|} \times \text{Entropy}(S_v)$$

Random Forest A popular ensemble learning technique for classification and regression applications is random forest. Building a sizable collection of decision trees, each trained on a different portion of the dataset, is how it works. Based on a distinct sampling of the data (a technique called bootstrap sampling) and a random subset of the attributes taken into account at each split within the tree, each tree in this ensemble provides an individual forecast. Each tree is distinct due to this unpredictability at the data and feature levels, which lowers the possibility of over fitting and increases the model's overall accuracy by lowering correlation between the trees.

Each tree "votes" on the anticipated class in the classification setting, and the Random Forest uses the majority vote to determine the final prediction. The final result for regression tasks is calculated by averaging the predictions made by each tree. Random Forest measures the "purity" of nodes in classification tasks using entropy or gini impurity to construct each decision tree. For example, entropy measures the degree of disorder in a node, whereas gini impurity determines the probability that a sample would be incorrectly classified if it were randomly allocated a class based on the distribution in that node. The formula for gini impurity is defined as:

$$Gini = 1 - \sum_{i=1}^{C} p_i^2$$

Mean squared error, which calculates the deviation between forecasts and actual values within each node, is frequently used for regression tasks with the goal of minimizing this error through split creation. The formula is:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

prediction using Random Forest majority voting formula:

$$\hat{y} = \text{argmax}_c \left(\sum_{k=1}^{K} \mathbb{I}(\hat{y}_k = c) \right)$$

This model has a number of advantages. Random Forest is more robust than single decision trees because it lowers the chance of over fitting by averaging predictions over several trees. In order to determine which features are most important for the model, Random Forest can also rank features according to how much they lower impurity across all trees. Despite its versatility and resilience, Random Forest can be computationally demanding, particularly when dealing with high-dimensional datasets or a large number of trees. Additionally, because of the complexity of having numerous trees, Random Forest is usually less interpretable than single decision trees.

Support Vector Machine It is a supervised machine learning technique that is mainly employed for classification tasks. Finding a hyperplane in a high-dimensional space that optimally divides the data into classes while maximizing the margin—the distance between the nearest data points from each class and the hyperplane—is the fundamental concept of support vector machines (SVM). These nearest points, referred to as "support vectors," are the most important informational points for identifying the best decision boundary. By addressing an optimization problem in which the objective is to maximize the margin while decreasing classification mistakes, the SVM method aims to create this hyperplane.

In SVM, prediction aggregation is the process of integrating the separate predictions of several support vectors to get a final prediction for a new data point. Finding the optimum hyperplane (also known as a decision border) to divide classes in a feature space is how it method operates. This hyperplane is defined in large part by each support vector, which is a data point that is closest to the decision border. The formula is defined as:

$$f(x) = \text{sign} \left(\sum_{i=1}^{l} \alpha_i y_i K(x, x_i) + b \right)$$

Nonetheless, a number of techniques can be used to expand SVMs to manage more than two classes in multi-class classification tasks. The One-vs-One (OvO) and One-vs-Rest (OvR) techniques are the most widely used strategies. A distinct binary SVM is trained for every class in the One-vs-Rest (OvR) technique, and each classifier separates one class from every other class. For each pair of classes in the One-vs-One (OvO) approach, a binary SVM is learned. Both approaches are predicated on the core ideas of support vector machines (SVM), which determine the decision boundary by optimizing the margin between classes and base the final prediction for a new data point on the sum of the output from each classifier. To ensure that each classifier adequately separates the classes, multi-class SVM extends the kernel function and decision boundaries to accommodate the many classes. OvO can occasionally provide better classification performance, particularly in situations with clearly defined decision boundaries, even if OvR is typically simpler and more effective in terms of the number of classifiers.

Long Short-Term Memory (LSTM) It is an RNN architecture designed to overcome typical constraints in capturing and learning long-term dependencies in sequential input. It is particularly effective in jobs involving time series data, NLP, speech recognition, and other similar tasks. LSTM is frequently implemented in practical applications using deep learning frameworks such as tensor-flow, Keras, Pytorch, and others, where developers and researchers can easily build, train, and deploy LSTM models for a variety of purposes such as time series prediction, language translation, sentiment analysis, and more. The LSTM gate equations are:

$$\begin{aligned} i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\ f_t &= \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \\ g_t &= \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \\ o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\ c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \tag{4.1}$$

This code represents the equations for input gate (i_t), forget gate (f_t), cell state (c_t), and output gate (o_t) of an LSTM cell. In these equations, h_t represents the hidden state at time step t, c_t represents the cell state at time step t, x_t represents the input at time step t, W_f, W_i, W_g, W_o are weight matrices for the forget gate, input gate, input modulation gate, and output gate respectively, and b_f, b_i, b_g, b_o are bias vectors for the corresponding gates.

For multi-class classification applications involving sequential data, including text, audio, or time-series analysis, they are ideally suited. This model updates its internal memory and hidden states at each time point as it processes input sequences in a multi-class classification problem. The input, forget, and output gates and memory cells that make up the architecture enable it to efficiently retain or discard data over time and capture long-term dependencies. A fully connected (dense) layer is applied to the model's final hidden state after the entire sequence has been processed, yielding a set of raw output scores. The softmax activation function is then used to normalize these scores, turning them into summable probabilities, each of which represents the likelihood that the input sequence falls into a particular class. The class with the highest probability is chosen as the final prediction. Even in applications with lengthy sequences or complicated dependencies,they can accurately classify data because of their exceptional ability to handle complex temporal patterns. The softmax function is shown by the equation

$$P(y = k \mid x) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}$$

This model have a number of benefits, especially when managing sequential data. They successfully overcome the vanishing gradient issue that plagues conventional RNN and capture long-term interdependence. This makes them perfect for tasks like speech recognition, natural language processing, and time-series forecasting that call for the learning of intricate temporal patterns. They can adjust to data with complex, dynamic relationships and are resilient to different sequence lengths. They are also very adaptable and accurate for sequential and multi-class classification issues because of their capacity to selectively store, forget, and output information utilizing memory cells, which improves their performance in tasks

requiring long- and short-term dependencies.

4.3.0.1 Cyber-Threat Attribution

The next step is cyber-threat attribution, which involves determining who is responsible for a cyber-attack. The use of machine learning in cyber-threat attribution has grown in recent years. It is becoming increasingly significant and is also a pressing requirement. Accurate defense strategies are required to safeguard organizations from future attacks. It is required to diversify a method that detects attacks with more Precision to make an informed decision on attack detection, Despite significant efforts by researchers, cyber-threat attribution continues to encounter hurdles in improving detection accuracy, necessitating the development of methodologies to better characterize attribution. Many researchers used ML techniques to evaluate their performance. Single, hybrid, and ensemble ml algorithms are used. They will aid in precise prediction. The dataset created in the previous phase is used for attribution in this phase.

The notions of testing and training datasets are fundamental to the process of developing and evaluating predictive models in ml. Typically, the dataset is separated into two subsets: the training and testing dataset. The training dataset is a subset of the whole dataset used to train the ML model. It uses this dataset to learn the patterns, relationships, and characteristics in the data. To create predictions or classifications, the model modifies its parameters and learns from the training data. The testing dataset is a subset of the whole that is not used during training. After training the dataset, the model is evaluated on the testing to determine its performance and generalization capabilities. The testing dataset assists in estimating how well the model will perform on new, previously unseen data. The separation of the dataset into training and testing sets is critical for appropriately evaluating the performance of the model.

In practice, a third subset known as the validation dataset is also frequently utilized. Various ML algorithms are used during this step. Because there are more than two classes, this is a multi-class classification problem. The number of CTAs in this investigation is twelve. The proposed study employs the categorization

techniques decision tree, random forest, and SVM. These algorithms were chosen because they perform well on textual data.

4.4 Behavioral Feature Attribution

Behavioral feature extraction methodology in cyber-threat attribution is a critical procedure that involves identifying and characterizing the behavioral patterns and properties of cyber threats. This methodology seeks to analyze and extract essential behavioral characteristics from various cyber-attacks, allowing cyber-security specialists to attribute these attacks to specific threat actors, groups, or entities.

Organizations may strengthen their threat-intelligence skills and overall security posture by understanding the distinct behavioral traits associated with different cyber threats. The proposed methodology consists of the three steps listed below. The phase of data collection comes first. During this stage, data is obtained. The second level is feature analysis. This step extracts features after doing the necessary text pre-processing. The third stage is threat attribution.

4.4.1 Proposed Framework for Behavioral Features

The framework is depicted in figure 4.6. The initial stage of data pre-processing is the collection of a dataset for experimentation. Stop words and special characters are removed using NLP approaches at this level. Following that, the text is cleaned up using methods such as lowercase conversion, tokenization, and lemmatization. The next step is the feature analysis phase. Word embedding models such as CBOW and skip-gram are used at this level.

For feature extraction, a novel embedding approach based on domain-specific data named attack2vec [190] is used. The data is separated into training and testing data in the third stage, CTA attribution. The data used for training and testing is divided in an 80:20 ratio.

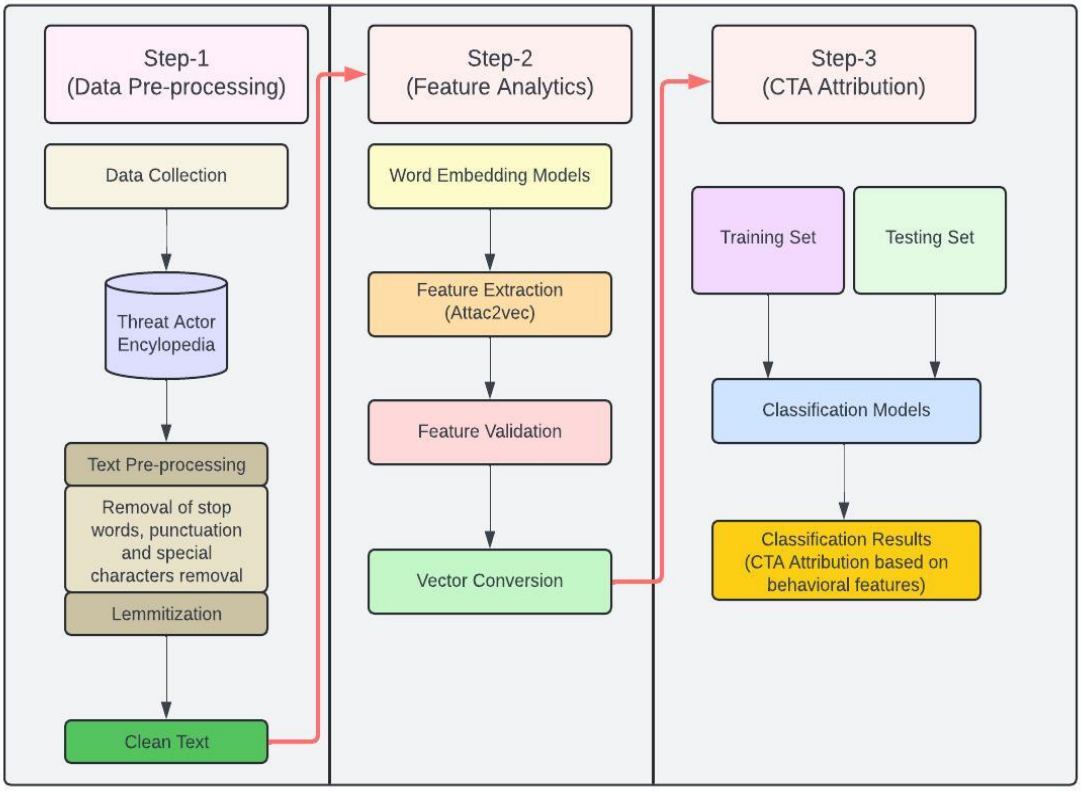


FIGURE 4.6: Proposed Framework for Behavioral Features.

4.4.2 Data Flow for Behavioral Features

A data flow diagram for the proposed methodology is shown in figure 4.7. The dataset used as input in this flow diagram is the threat actor encyclopedia. Text pre-processing is the next step. Following this stage, data is gathered to extract behavioral features. The classification step, which follows the extraction of behavioral data, involves a range of machine/deep learning approaches.

4.4.3 Phases for Extraction of Behavioral Features

4.4.3.1 Data Gathering

The initial stage is to gather essential data on the cyber-attacks under investigation. This data may comprise network traffic logs, system event logs, malware

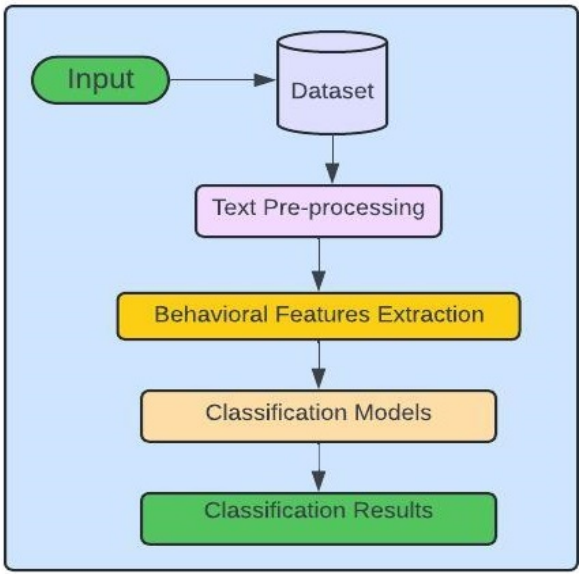


FIGURE 4.7: Flow Diagram for Behavioral Features.

samples, attack patterns, and any other relevant information that can provide insights into how cyber threats behave. For technical considerations, datasets in the form of unorganized CTI reports are accessible for information extraction. The availability of datasets remains a challenge in this domain for extracting behavioral features. So yet, the research has not examined their impact. Because attack methods are always evolving, it is critical to identify cyber threat actors based on behavioral characteristics. Incorporating these traits into this domain will aid in locating the context of threat actors.

In this regard, the threat-actor encyclopedia is a dataset provided by Thai Cert in 2019. This dataset investigates the attacker's behavioral traits. This data collection indicates the attacker's intentions and reasons behind an attack.

4.4.3.2 Text Pre-processing

After collecting the data, it must be pre-processed to remove any noise or useless information. This stage entails cleaning, normalization, and transformation of data to ensure that it is in a format appropriate for future analysis. In this phase, the text was cleaned using several NLP methods, as shown in figure 4.8 (removal of stop words, punctuation, and lemmatization). The text is first changed to

lowercase. The text is stripped of stop words, punctuation, and special characters. Tokenization is then carried out. Finally, the text is trimmed using lemmatization.

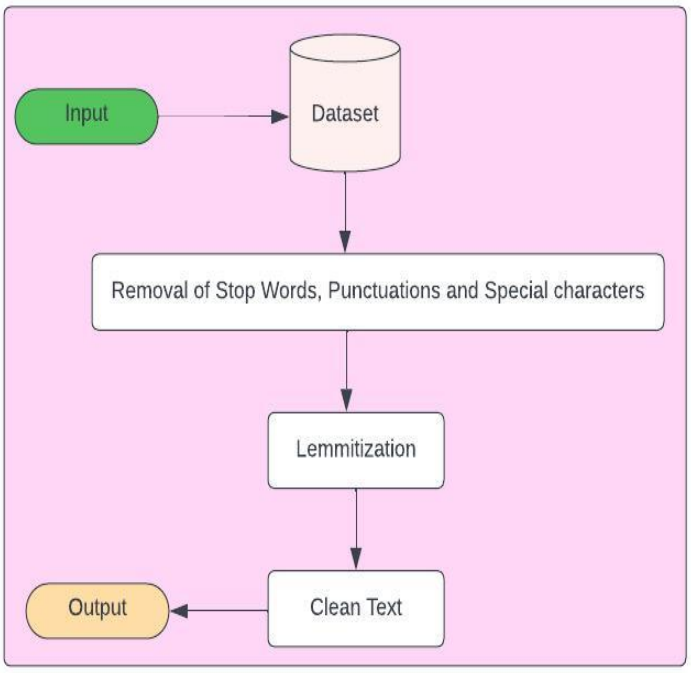


FIGURE 4.8: Text Pre-processing.

4.4.3.3 Feature Analytic

In the context of data analysis and ML, feature analysis refers to the process of studying and comprehending the significance and influence of numerous characteristics or variables inside a dataset. The measurable aspects of data that are used as inputs for an ML model to create predictions or classifications are referred to as features. Feature analytics entails investigating the correlations and patterns that exist between these features to obtain insight into their individual significance and collective influence on the model's prediction performance. The second phase is the extraction of features from the dataset.

Data scientists and analysts can use feature analysis to improve the performance of machine learning models, improve predicted accuracy, minimize computational costs, and obtain a better understanding of the underlying data patterns. This

method is vital for developing robust and efficient models, particularly in complicated real-world applications where feature quality and relevance are critical to the model's predictive accuracy and generalization capabilities.

4.4.3.4 Feature Extraction

The collected data is then subjected to feature extraction algorithms to gain useful insights and identify significant behavioral patterns linked with cyber threats. This may entail extracting relevant behavioral features from a dataset using statistical analysis, ML algorithms, or other data mining techniques. The data set's features are extracted at this stage. For this, the innovative embedding model "attack2vec" is used. This model was created by combining domain-specific embedding and cyber-threat intelligence. Model performance is determined using models such as skip-gram and CBOW. They are the most commonly used in the literature and are regarded as the best in this discipline. In the proposed work, these two models are evaluated using a variety of classifiers, including decision trees, random forests, support vector machines, and the deep learning classifier long short-term memory (LSTM).

4.4.3.5 Vector Conversion

Converting text into vectors is a fundamental process in NLP that allows machines to understand and process textual data. The choice of vectorization technique depends on the task at hand and the nature of text data being processed. Vector embedding is one of the most exciting methods. Many NLP recommendations and search algorithms rely on them. Vector embedding is a critical component of NLP.

These models may efficiently handle and analyze textual data for diverse tasks such as querying, classification, and sentiment analysis by representing words or phrases as high-dimensional mathematical vectors. While the concept of vectors applies to many disciplines, it is critical to recognize that mathematical vectors are

not constrained by the dimensions of the physical world. In this phase, text data is converted into vectors for effective utilization of machine learning algorithms.

4.4.3.6 Threat Attribution

The goal of this step is to identify the cyber-threat actor attack patterns. During this stage, the dataset is mined for behavioral elements. This phase employs a variety of ML and deep learning algorithms, including decision trees, random forests, and LSTM. The usage of decision trees allows for the creation of subset characteristics at various phases of development.

Furthermore, data scaling and normalization are not required. A supervised machine learning classifier known as the random forest is made up of several decision trees. It generates a smaller data collection, which improves classification accuracy.

The bagging model enhances the classifier's performance. It outperforms decision trees in most instances. Even when there are missing values in the data collection, the findings are still satisfactory. Even without hyper-parameter adjustment, it can yield good results. It also overcomes the problem of over fitting in decision trees.

LSTM, a deep-learning recurrent neural network, is used for text classification. LSTM surpasses recurrent neural networks in terms of performance because it overcomes the issue of long-term dependency. The LSTM architecture employed in this study is depicted in figure 4.9.

It is made up of three layers: the input layer, the concealed layer, and the output layer. Threat attribution employs both behavioral and hybrid aspects.

Table 4.2 displays the extracted features. Motivation, first observed, operations executed, sponsor by, origin nation, outcome, and attacker skill are the behavioral features derived from the dataset.

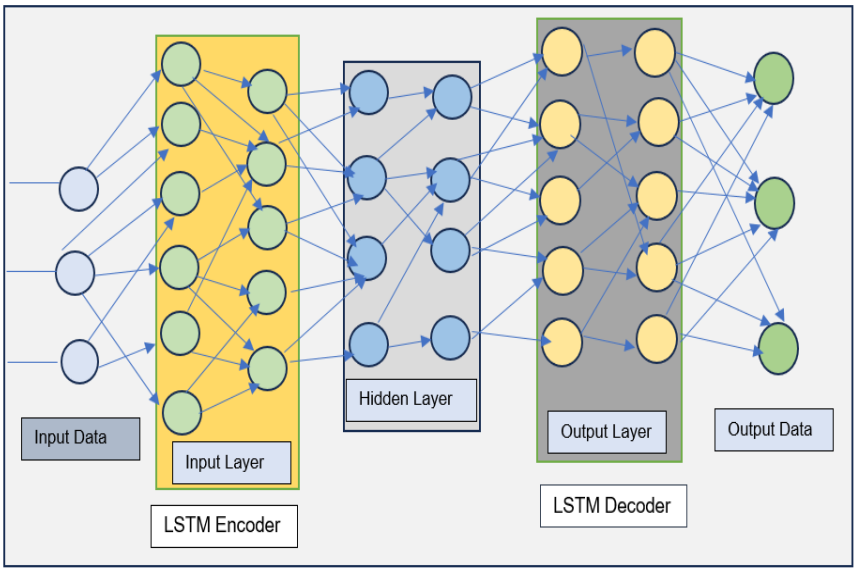


FIGURE 4.9: LSTM Architecture for Attack Attribution.

TABLE 4.2: Behavioral Features in Attack Attribution.

Sr. #	Features used
1.	Motivation
2.	First seen
3.	Operations performed
4.	Sponsor by
5.	Origin country
6.	Outcome
7.	Attacker skill

4.5 Hybrid Feature Attribution

To obtain overall knowledge of feature relevance, hybrid attribution may involve combining and exploiting the capabilities of these distinct methodologies. The combined effect of hybrid features is evaluated to analyze the impact on attribution process.

To determine the influence, hybrid features are merged in this stage. These hybrid traits employed are shown in table 4.3. The proposed framework for hybrid

TABLE 4.3: Hybrid Features.

Sr. #	Features
1.	TTP
2.	Malware
3.	Tools
4.	Target organization
5.	Target country
6.	Target application
7.	Motivation
8.	First seen
9.	Operations performed
10.	Sponsor by
11.	Origin country
12.	Outcome
13.	Attacker skill

features is shown in figure 4.10.

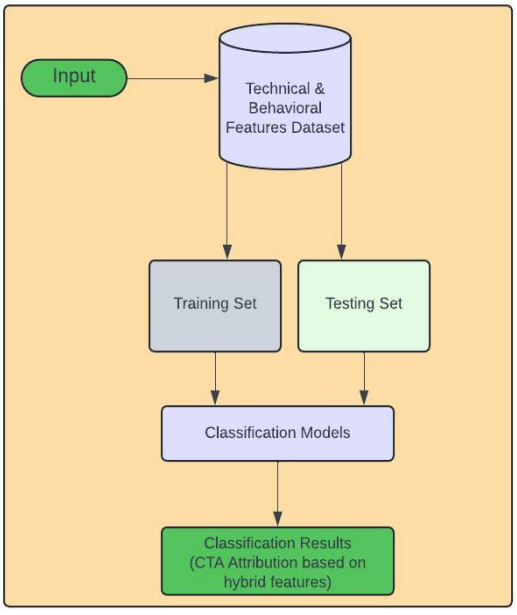


FIGURE 4.10: Proposed Framework for Hybrid Features.

4.6 Optimal Feature Selection

There is a need to select optimal features for cyber-threat actor attribution because not all of hybrid features do not carry the same level of relevance or predictive power for identifying specific threat actors. Not every feature in the 13-set will contribute equally to distinguishing between different threat actors, and some may be redundant or less relevant to the specific patterns of behavior associated with an attacker. By selecting optimal features, the model focuses on those that provide the highest relevance. By utilizing the most important variables—which are essential for differentiating between threat actors this feature selection process improves the models performance and makes it easier to understand. As a result, choosing the best features contributes to increasing the precision and effectiveness of cyber-threat attribution, making it a more trustworthy method of locating attackers in intricate, real-world situations.

The process of selecting a subset of relevant features or variables from a broader collection of available features in a dataset is referred to as feature selection methodology. The goal is to discover and maintain the most critical and useful properties while removing irrelevant, redundant, or noisy. This is a technique that is extensively used in machine learning and data analysis to improve model performance, reduce over-fitting, and increase computing efficiency. To begin, understand the problem domain and the facts at hand. Determine the goal of feature selection—whether it is to reduce dimensionality, improve model accuracy, or simplify model interpretation. The proposed framework for optimal features is shown in figure 4.11.

4.6.1 Need of Feature Selection

Choosing the best features for ml and data analysis is crucial. It entails selecting the most relevant features from a dataset while rejecting unnecessary, redundant, or noisy ones. This procedure is critical for increasing model performance, decreasing over-fitting, and improving model interoperability. Feature selection

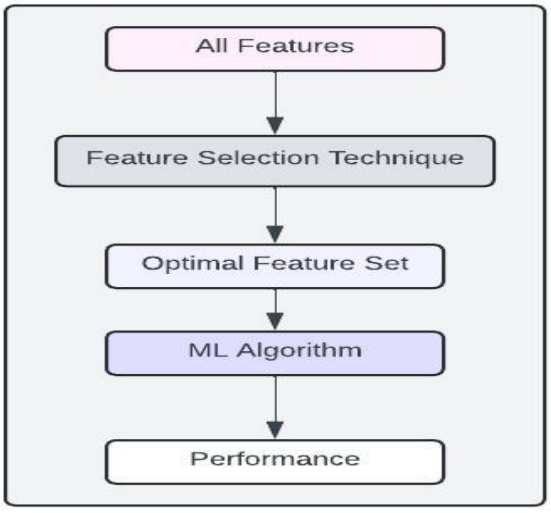


FIGURE 4.11: Proposed Framework for Optimal Features.

techniques are crucial in machine learning and data analysis for several reasons. Unnecessary or duplicated information is avoided by picking the most relevant features. Avoiding over-fitting and enhancing generalization to new data, can lead to more accurate and efficient models. This is especially useful for large datasets or sophisticated models.

4.6.2 Process of Feature Selection

Understanding the major factors that influence the models predictions is aided by feature selection. It simplifies the model by emphasizing the most critical aspects, making it easier to explain and interpret. Models that are smaller and more streamlined are easier to deploy, maintain, and update particularly in real-time or resource-constrained contexts. In high-dimensional datasets, feature selection aids in mitigating the curse of dimensionality by enhancing model performance and lowering the chance of errors or inefficiencies. Feature selection can direct these efforts by emphasizing the most important attributes; hence reducing time and effort in the feature generation process. These advantages add up to more efficient, accurate, and interoperable machine learning models, making feature selection techniques an essential part of data preparation and model construction in

a variety of areas.

The purpose is to find the optimal feature set for attribution. Hybrid features are used in this technique to select the best one for threat attribution. Among various feature selection procedures are filter-based, wrapper-based, embedding methods, and hybrid approaches. Filter-based procedures include the chi-square test, the Fisher score, the correlation coefficient, the variance threshold, and the mean absolute difference. Wrappers are used to implement forward, backward, exhaustive, and recursive feature selection algorithms. LASSO and random forest regularization are examples of embedded approaches. Each methodology has advantages and disadvantages, and the approach chosen typically depends on the nature of the problem, the dataset, and the machine learning algorithm utilized. To develop an efficient and accurate model, a compromise between lowering dimensionality and maintaining significant information must be struck.

4.6.3 Genetic Algorithm

The genetic algorithm is also employed as a feature selection tool at this level. Genetic algorithms excel at optimizing complicated landscapes with many variables, particularly when standard methods fail because of the large search space. They are particularly effective in high-dimensional and non-linear optimization issues. It excels at optimizing complicated landscapes with many variables, particularly when standard methods fail due to the large search space. They are particularly effective in high-dimensional and non-linear optimization issues. When compared to other techniques, it is comparatively easy to implement and understand.

It includes basic genetics-inspired procedures like selection, crossover, and mutation, making them easy to install and experiment with. The process of the genetic algorithm is depicted in figure 4.12. In this study, feature selection for cyber-threat attribution was optimized using a genetic algorithm (GA). Classification accuracy was used as the fitness function to evaluate the performance of each of the initialized 30 candidate feature subsets in the population.To increase attribution accuracy, the GA iteratively refined these subgroups over a 20-generation.

The idea of crossover is adaptable and has uses in a variety of industries. It describes how chromosomes exchange genetic material during meiosis, which contributes to genetic variety in progeny, in the field of genetics. With a crossover rate of 0.8, 80% of the population experienced crossover, allowing features from different subgroups to recombine and produce new candidates. Furthermore, controlled randomness was provided with a mutation rate of 0.05, preserving population diversity and avoiding convergence on less-than-ideal solutions.

The model's capacity to correctly categorize cyber-threat actors was improved by the extraction of a hybrid collection of attributes made possible by this method. In this phase, the genetic algorithm is used to pick features. It outperforms other approaches to feature selection. One advantage is that it works better and gives higher-quality results with larger data sets.

After applying feature selection techniques to a total of 13 features (TTP, tools, malware, target country, target organization, application, motivation, first seen, operations performed, sponsor by and origin country), seven features TTP, tools, malware, motivation, sponsor by, outcome and attacker skill are chosen. These are regarded as the most appropriate and optimum characteristics for cyber-threat attribution. The generation view is shown in figure 4.13.

4.7 Attack Detection in IDS

To successfully identify and respond to potential threats or assaults in a network, an IDS employs a combination of tactics, strategies, and technologies. Define the sorts of threats to be detected, as well as the scope and criteria for evaluating the IDS performance. Use cyber-threat intelligence insights to improve IDS signatures. These are patterns or specific IoC used to detect known threats. Mapping threat intelligence data to IDS signatures allows to construct more precise and tailored detection rules. Integrate behavioral analysis techniques into IDS to detect aberrant behavior that could indicate a cyber-attack. Cyber-threat information can help us determine what constitutes aberrant behavior based on recognized threat

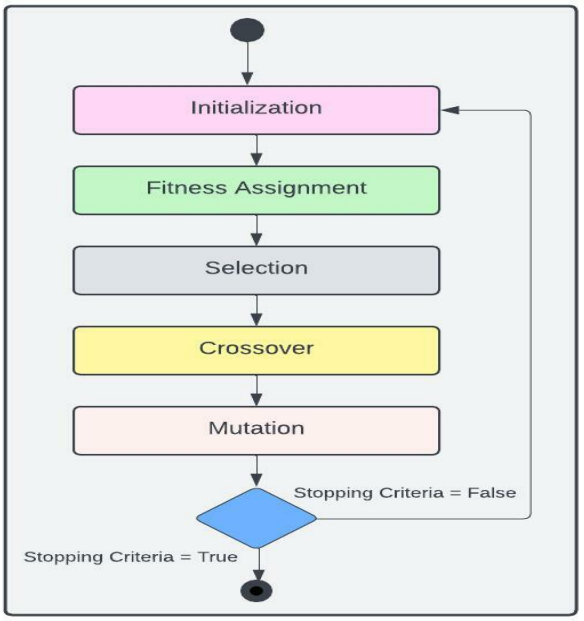


FIGURE 4.12: Genetic Algorithm Process of Feature Selection.

gen	nevals	avg	std	min	max
0	100	0.39	0.143571	0.1	0.575
1	53	0.5015	0.0599812	0.3	0.575
2	63	0.53575	0.0735914	0.2	0.6
3	68	0.56575	0.0416316	0.275	0.6
4	51	0.5765	0.026415	0.45	0.6
5	60	0.57725	0.0552376	0.2	0.6
6	57	0.578	0.0694874	0.275	0.6
7	66	0.564	0.0967161	0.275	0.6
8	54	0.57825	0.0719944	0.275	0.6
9	62	0.593	0.0247689	0.425	0.6
10	57	0.57225	0.0826056	0.275	0.6

FIGURE 4.13: Generation View of Optimized Feature Selection.

actors' TTP. Unstructured CTI reports provide useful information on emerging threats, attack patterns, and adversary behaviors. While unstructured data can be difficult to analyze, natural language processing (NLP) techniques can be used to extract essential elements and put them into a structured format appropriate for IDS analysis. The organized data can then be utilized to create or enhance IDS signatures. Customized signatures and rules based on CTI behavioral characteristics can be developed. These signatures and rules are designed to identify IoC linked with known threat actors, such as IP addresses, domain names, file hashes,

and network traffic patterns. Threat intelligence inputs from CTI sources can be integrated into IDS infrastructure. These feeds offer real-time or near-real-time updates on emerging threats, such as new TTP, malware variants, and targeted vulnerabilities.
Security events detected by IDS can be co-related to attribute data from CTI sources. By analyzing the context provided by CTI, such as assigning an attack to a specific threat actor group, security analysts can better prioritize and examine IDS warnings. CTI data can be used to improve the contextual analysis capabilities of IDS.

By incorporating information about threat actor goals, sponsorship's, and geopolitical affiliations, IDS can gain a better understanding of the larger context of identified security events and analyse their possible impact on organization. The suggested methodology examines two data sets: NSL-KDD and CSE-CIC-IDS2018. These are the two most utilized data sets in IDS analysis of attack detection.

The study concentrated only on the theoretical and computational elements of integrating CTI with machine learning models for attack detection, without delving into the actual implementation or modification of IDS tools and accompanying rule sets.

4.8 Methodology for NSL-KDD Dataset

There are three stages to the suggested methodology for analyzing the NSL-KDD dataset. In the first stage, data transformation techniques are used. The second phase is the reduction of features. The third phase is using classification methods like SVM, random forest, and decision tree.

The proposed technique for the NSL-KDD dataset is shown in figure 4.14. The proposed methodology is divided into three stages. The initial stage is data preprocessing. At this stage, the dataset is translated into numerical values using data transformation techniques such as label encoder. Because ML algorithms perform

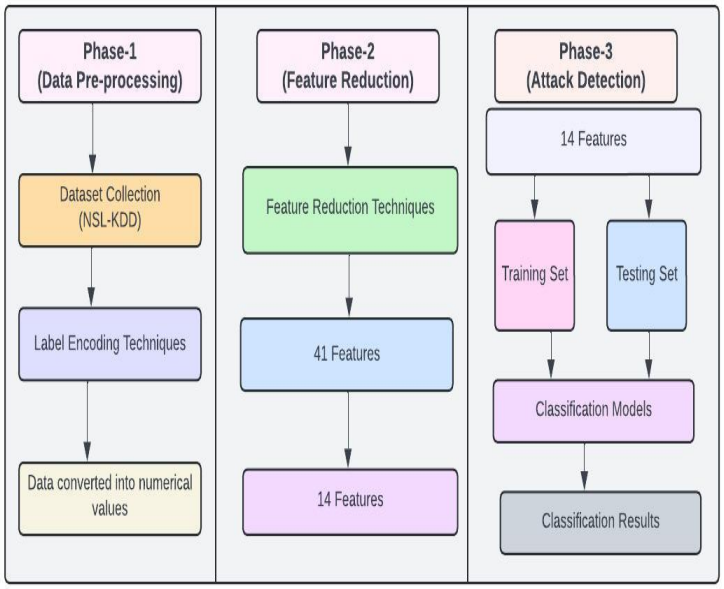


FIGURE 4.14: Proposed Methodology for NSL-KDD Dataset

best on single-value datasets, data transformation techniques are employed to convert the dataset to a single numerical value. The second stage is feature reduction. Techniques such as PCA are used to minimize the feature set.

4.8.0.1 Data Transformation Phase

The NSL-KDD dataset consists of both numerical and nominal values. All are converted to numerical in this phase. Transformation in the context of machine learning refers to the process of transforming or converting data from its original format into a different format that is more suitable for analysis, model training, or downstream processes.

Data transformation is an important stage in the machine learning pipeline since it improves data quality and makes it more compatible with the algorithms or models being employed. These transformations are carried out to improve the quality of data, reduce noise, and ensure that the machine-learning model can learn effectively from the provided data. These transformations are carried out to improve the quality of data, reduce noise, and ensure that the machine-learning model can learn effectively from the provided data. It is critical to the success of machine

learning models because the quality of input data has a substantial impact on the performance and accuracy of the models trained on it. Using a label encoder for this transformation is employed since it is the most widely used method. Converting values to a single value has the advantage of generating correct results because machine learning algorithms work well on single types of values.

Label encoding is simple and easy to implement. It gives each category a unique numerical value, making it easy to work with categorical data in numerical form. It saves the information included in the categories. Each category is given a unique numerical value that can be used to represent it in computations. Numerical inputs are required by many machine learning algorithms and libraries.

Label encoding is very useful when utilizing algorithms that are designed to work with numbers like decision trees or random forests. Label encoding often has a minimal computational cost because it simply substitutes categorical variables with numeric values, which eliminates the need for considerable computer resources.

4.8.0.2 Feature Reduction Phase

Feature reduction, also known as dimensionality reduction, is an important stage in machine learning and data analysis, especially when working with datasets with many variables or features. The basic goals of feature reduction are to simplify the model by minimizing the number of features while maintaining performance, to reduce the computational load and time necessary for model training, and to reduce the risk of over-fitting, redundant, or irrelevant features being removed.

The goal of this phase is to save processing power because the dataset has forty-one features that demand more processing power to compute the values. Several feature reduction methods are used in the literature, including genetic algorithms, linear discriminant analysis (LDA), principal component analysis (PCA), information gain, and generalized discriminant analysis (GDA). PCA is the feature reduction method that is currently most widely utilized around the world. PCA is used in this case because it is easy to calculate and produces accurate results.

TABLE 4.4: Optimal Features for NSL-KDD Dataset.

Sr. #	Features
1.	Duration
2.	Protocol_type
3.	Src_bytes
4.	Dst_bytes
5.	logged_in
6.	Count
7.	Service
8.	Num_failed_logins
9.	Error_rate
10.	Root_shell
11.	Serror_rate
12.	Dst_srv_rate
13.	Hot
14	Is_guest_login

Problem-solving is straightforward for computing systems. Lowering dimensionality improves the performance of machine learning algorithms. PCA has the advantage of reducing data noise. The genetic algorithm, for example, has a significant computing cost. Data with large dimensions is difficult to visualize; so, PCA simplifies data visualization by reducing the dimension. The proposed study's feature set consists of 41 features. The initial forty-one set is reduced using PCA, and the fourteen best features are picked. A threshold is specified, and values more than 0.60 are considered features.

In this sense, fourteen feature sets have been chosen. These approaches have the advantage of speeding up the system and utilizing fewer processing resources by reducing the amount of data set features. Table 4.4 shows the ideal 14-feature set derived via PCA.

4.8.0.3 Classification Phase

The next phase is applying a classification algorithm on the data extracted from phase 2 with fourteen features. For classification, the SVM, RF, and DT are utilized. Figure 4.15 displays a flow diagram. The NSL-KDD data set serves as the system's input. Using data transformation techniques, the data is reduced to a single numerical value. The features in the data set are then reduced using feature reduction techniques. To distinguish between legitimate and malicious traffic, classification algorithms are used after feature reduction procedures.

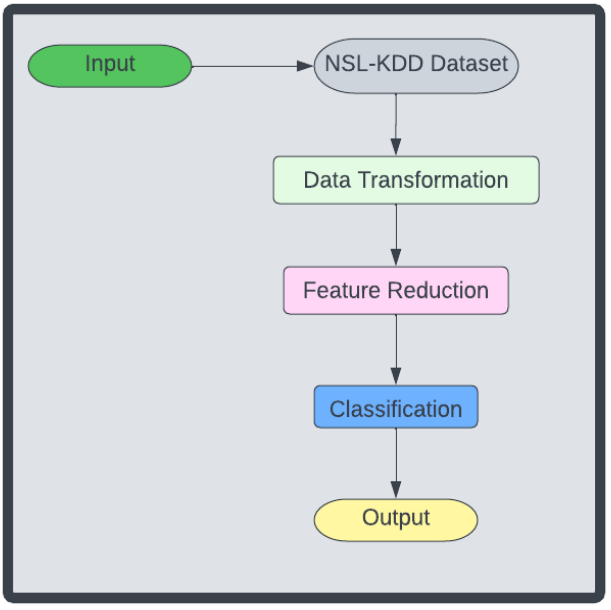


FIGURE 4.15: Flow Diagram-NSL-KDD Dataset

In this phase, 41 characteristics are reduced to fourteen. When more features are used in the dataset, computational power increases. As a result, feature reduction techniques are used to conserve computational resources.

The third stage involves using machine learning methods for classification. In this step, the decision tree, random forest, and SVM algorithms are used to categorize data. The datasets for training and testing are split 80:20. Machine learning techniques identify whether the data is an attack or legitimate/normal traffic.

4.9 Methodology for CSE-CIC-IDS2018 Dataset

The CSE-CIC-IDS2018 dataset analysis is divided into three steps. The first stage is the normalization phase, which includes approaches such as z-score and min max. The second phase employs feature reduction techniques such as PCA, while the third employs classification methods such as SVM, RF, and DT.

The proposed technique for the CIC-IDS2018 data set is shown in figure 4.16. The proposed methodology is divided into three stages. The first stage is the normalization phase. In it, the data set is normalized using normalization techniques such as z-score.

Normalization is a widely used method for preparing data machine learning. The process of transforming numeric column values in a dataset to a standard scale while preserving information and not distorting the value ranges is known as formalization. The second stage consists of reduction. Techniques such as PCA are used to minimize the feature set at this phase. The third phase is classification.

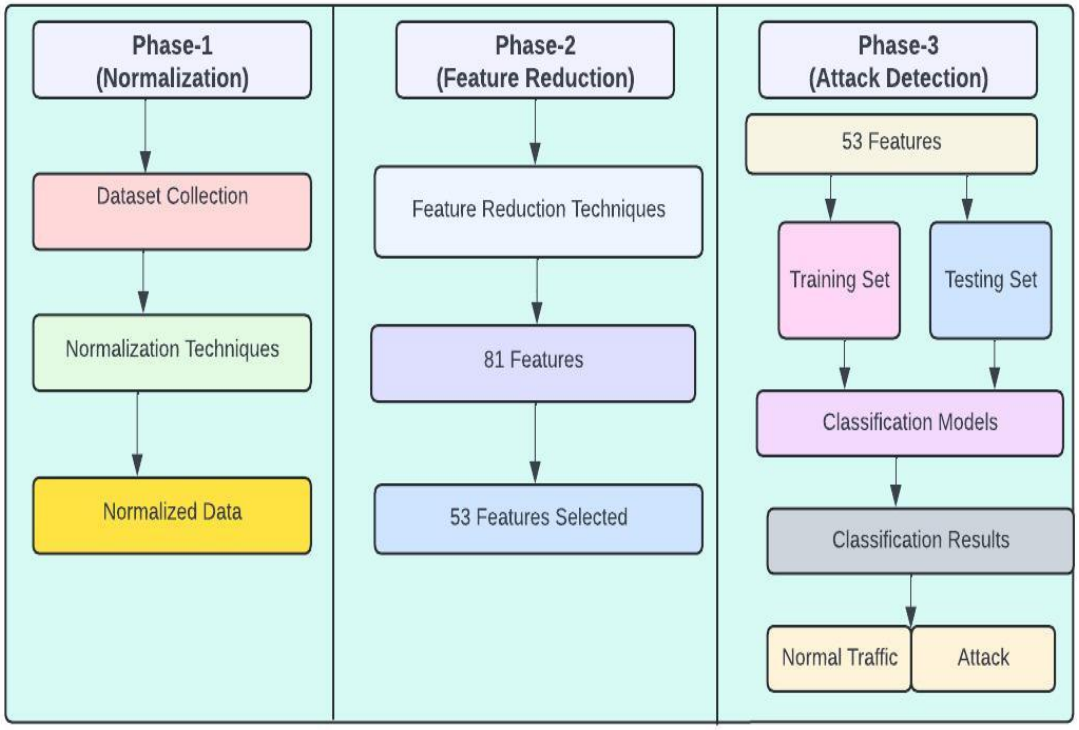


FIGURE 4.16: Proposed Methodology of CIC-IDS2018 Dataset

4.9.0.1 Normalization Phase

The first step is to standardize the data. Because values in several columns of datasets are pretty high. To balance the values in data, normalization procedures are applied. The benefit of adopting these techniques is that it equalizes all of the column values. Z-score is utilized for this purpose. It is also known as a standard score, which is a statistical metric that defines the relationship of a value to the mean of a collection of values in terms of standard deviations from the mean.

It is a method of standardizing disparate datasets so that they may be compared. The Z-score reveals how far a data point deviates from the mean. A Z-score of zero indicates that the data point is exactly at the mean. A positive z-score indicates that the data point is above the mean, whereas a negative z-score indicates that it is below the mean.
The greater the deviation from zero, the more exceptional or extreme the data point is compared to the remainder of the dataset. It is especially useful for comparing diverse datasets with varied means and standard deviations, allowing for a standardized comparison. They aid in the identification of outliers, comprehending the relative position of a certain data point within a dataset, and permitting more meaningful comparisons between different sets of data.

4.9.0.2 Feature Reduction Phase

In the second phase, a normalized dataset is used for feature reduction because it has eighty-one features that demand more computational power and resources to employ. PCA is used here for reduction. In it, the co-variance of features is calculated to determine the subset. The formula of co-variance is:

$$\text{cov}(X_i, X_j) = \frac{1}{n-1} \sum_{k=1}^{n} (X_{ki} - \bar{X}_i)(X_{kj} - \bar{X}_j)$$

The cutoff is set at 0.60. Values greater than this criterion are chosen. A total of 81 feature sets are reduced to 53.

4.9.0.3 Classification Phase

The next phase is applying a classification algorithm on the data extracted from phase 2 with fourteen features. For classification, the SVM, RF, and DT are employed. Figure 4.17 displays a flow diagram. The CSE-CIC-IDS2018 dataset serves as the system input.

Using normalization techniques, the data is normalized. Then the dataset features are reduced using feature reduction techniques. To distinguish between legitimate and malicious traffic, classification algorithms are used after feature reduction procedures.

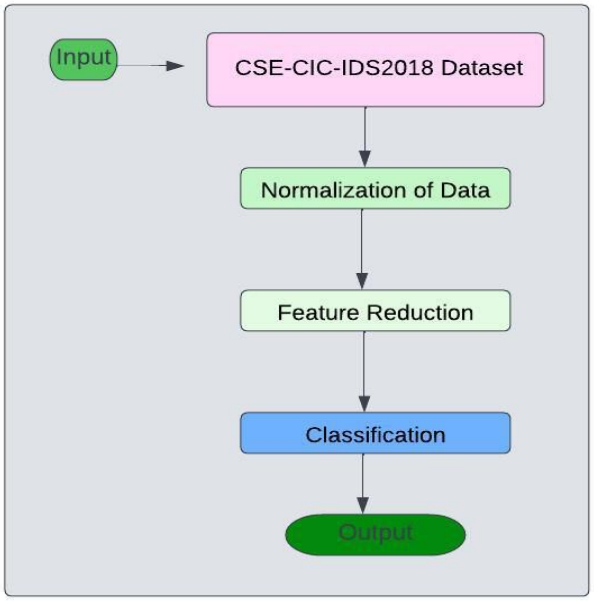


FIGURE 4.17: Flow Diagram CIC-IDS2018 Dataset

Here eighty-one qualities are reduced to fifty-three. The computing power of a data set grows as more features are used. As a result, feature reduction techniques are used to conserve computational resources.

The third stage involves using machine learning methods for classification. In this step, the decision tree, random forest, and SVM algorithms are used to categorize data. The datasets for training and testing are split 80:20. Machine learning techniques identify whether the data is an attack or legitimate/normal traffic.

4.10 Catering Zero-day Attacks

The methodology adopted in this research dissertation is well-suited to counter zero-day attacks. They use undiscovered vulnerabilities and frequently employed methods that have never been seen before. Thus making it challenging for systems that depend on known signatures to identify them. However, by concentrating on the methods that are typical of different actors and campaigns, this methodology makes use of optimal features acquired from previous threat actor behavior and attack patterns, enabling it to detect even zero-day attacks.

It creates a profile that captures the patterns of threat actors by utilizing both technical and behavioral features. Furthermore, the validation of features from the MITRE ATT&CK framework improves its resistance to zero-day attacks. With its extensive coverage, this framework provides a thorough benchmark for evaluating novel attack behaviors.

Finally, by utilizing the optimization of particular characteristics, this model can identify CTA in real-time, removing the need for extensive CTI data. This identification is essential for reacting early to the zero-day attack.

This approach allows for proactive defense against attackers who try to avoid detection by using creative exploits by establishing a detection model that recognizes threat actors based on widely applicable properties. Even in the case of zero-day attacks, this method efficiently attributes cyber-threat actors, enhancing organizational resilience.

4.11 Catering Fake Threat Advisories

This adapted model emphasis on key features like TTPs, tools, target country, and motivating taken from trustworthy threat actor profiles helps to reduce fake advisories, which are frequently created to deceive by inflating or inaccurately attributing assaults. It uses CTI reports from well known security vendors, thus

reducing the chance of fake advisories. Furthermore, by validating features against realistic, well-known threat frameworks like MITRE ATT&CK architecture helps to identify inconsistencies in the event that a threat advisory describes tactics that do not match actual adversary patterns. The model emphasis on optimal attributes increases its resistance to fake reports. The model can preserve accurate threat attribution by using behavioral consistency checks and focusing solely on the most pertinent features, weeding out possible false information from fake advisories.

4.12 Experimentation Methodology

Keeping in view our objective and skewed datasets, we adopted an experimental methodology that is used in various studies. Following this, we first perform several experiments to evaluate the results. Figure 4.18 illustrates the experimentation methodology. It demonstrates that gathering data is the initial step. For the same, a variety of datasets, including CTI reports, threat actor encyclopedia, NSL-KDD, and CIC-IDS2018, have been gathered.Following data collection, the threat-actor encyclopedia is used to carry out behavioral and technical feature attribution using the CTI reports that have been gathered. After that, the resulting dataset is used to perform hybrid feature attribution. To identify the optimum features for the attack attribution procedure, optimal feature selection is then carried out.

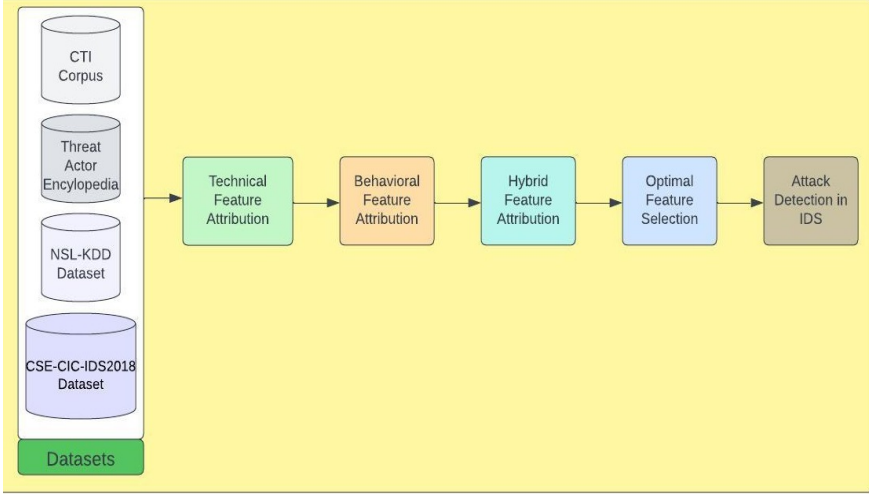


FIGURE 4.18: Experimentation Methodology

4.13 Summary

In this chapter, the methodology for cyber-attack attribution that incorporates both technical and behavioral elements is elaborated in detail. Initially, methods for attributing technical aspects are described, including a flowchart and a description of the text pre-processing stages. Domain-specific cyber-security embedding are used with classification algorithms such as decision tree, random forest, and support vector machine to analyze performance metrics such as Accuracy, Precision, Recall, and F1-measure.

The chapter then moves on to behavioral feature extraction, which includes text pre-processing and feature analytic. The suggested approach for this procedure is thoroughly developed, and performance measures are once again employed to evaluate the classification algorithms. The threat actor encyclopedia released by Thai CERT is the source of the behavioral characteristics employed in this model for cyber-threat attribution. Only these particular behavioral characteristics can be extracted due to the limitations of the dataset. Till now, there is no other dataset available for behavioral feature extraction. Hybrid features are employed by merging technical and behavioral ones, and their performance is analyzed similarly. Finally, the chapter discusses selection of optimal features for cyber attack attribution, which uses a genetic algorithm to improve the entire attribution process.

Chapter 5

Experimentation and Results

5.1 Introduction

Technical, behavioral, hybrid, optimal feature selection, and accurate attack detection in IDS methods have been evaluated in this chapter. Python is used for implementation. For performing experiments system used is a Core I-7 with 16 GB of RAM.

Embedding models are classified into two categories. Figure 5.1 depicts the CBOW with skip-gram. CBOW approach searches context words for the target word. Context word is sought from the appropriate target word in the skip-gram model. It is preferred where context is critical.

Both models were tested in this analysis. Different window sizes of n=3, 5, and 7 are employed. The associated vector size is 100. Skip-gram model outperforms CBOW in terms of results, hence it is favored over CBOW.

5.2 Results for Technical Feature Attribution

In this section results for technical features attribution were conducted to know the performance of various models. Firstly CBOW and skip-gram performance is evaluated. Then the final results for technical feature attribution are discussed.

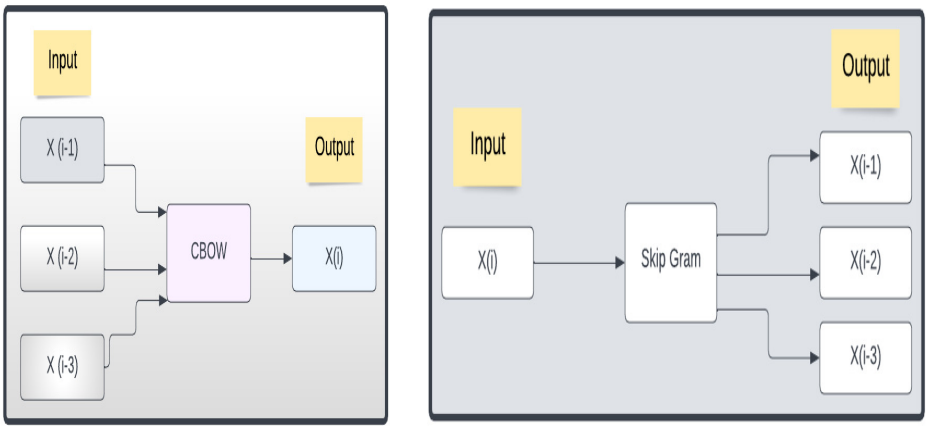


FIGURE 5.1: Comparison of CBOW and Skip-gram Model.

5.2.1 Model Performance

Embedding models with different machine learning algorithms such as decision trees, random forests & support vector machines are used. Performed various experiments with different window sizes of n=3, 5, and 7. The context window size (n) is the number of words before and after the target word that the model considers contextual. As the context window size increases, the model captures more distant word associations. With a smaller context window, the model can catch more local word associations.

It may excel at recognizing syntactic links and closely related concepts. However, it may struggle to capture distant semantic links. As the context window size grows, the model collects a wider range of contexts around each word. Tasks requiring an awareness of broader semantic linkages, such as word similarity or semantic relatedness, may increase performance. With a bigger context window size, the model captures more semantic links. Performance may be improved for tasks such as word analogies and document-level semantics. However, it may become more computationally expensive and memory-hungry.

With a window size of three, the CBOW model takes three context words on either side of the target word. Training loss lowers significantly over epochs as the algorithm improves its ability to predict target words. Validation loss is initially lower but may increase as the model begins to overfit the training data. The

model's accuracy on the training and validation sets may improve as it learns stronger word representations.

However, there may be a trade-off between computational efficiency and memory utilization. With a window size of 5, the CBOW model considers five context words on both sides of the target word. This wider context allows the model to capture more contextual information about the target word. In the performance graphs, there might be a smoother convergence of the training loss compared to n=3. The validation loss might be more stable, reflecting the model's ability to generalize better with a larger context window.

To summarize, as the window size rises, CBOW models may be able to capture greater contextual information, leading to improved performance in tasks such as word prediction, language modeling, and sentiment analysis. Accuracy might be slightly higher than in the n=3 case.

When CBOW is used with a decision tree for n=5, Accuracy, Precision, Recall, and F1-measure of 84%, 85%, 83.8%, and 84% are recorded. When CBOW is used with a random forest classifier, for n=5 we get Accuracy, Precision, Recall, and F1-measure of 87%,88.6%,87.1%, and 87%. With CBOW with SVM for n=5, Accuracy, Precision, Recall, and F1-measure of 85%, 83%, 85.9%, and 89.5% is recorded.

5.2.2 Skip Gram Model Performance

The performance of the skip-gram model is relatively better with different machine learning algorithms. When the decision tree is implemented, for n=5 highest Accuracy, Precision, Recall, and F1-measure of 85%, 83%, 85.9%, and 89.5% are seen. When skip-gram is used with random forest it produces the highest results overall. The Accuracy, Precision, Recall, and F1-measure are 96%, 96.6%, 95.58%, and 92% is recorded. When the skip-gram model is used with SVM, for n=5 highest Accuracy, Precision, Recall, and F1-measure of 89%, 91.2%, 91.5%, and 90.3% are recorded. The results of these experiments are shown in figure 5.2-5.7.

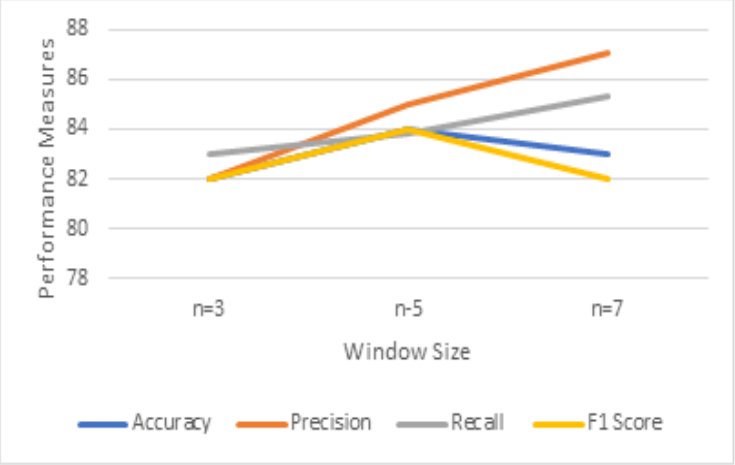


FIGURE 5.2: Performance of CBOW with Decision Tree.

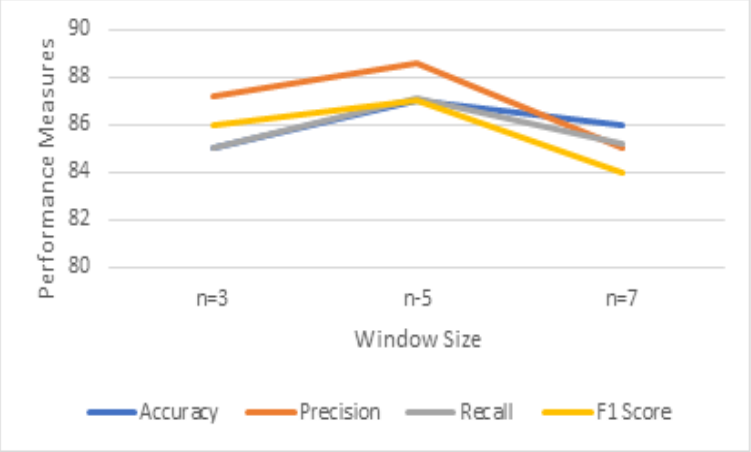


FIGURE 5.3: Performance of CBOW with Random Forest.

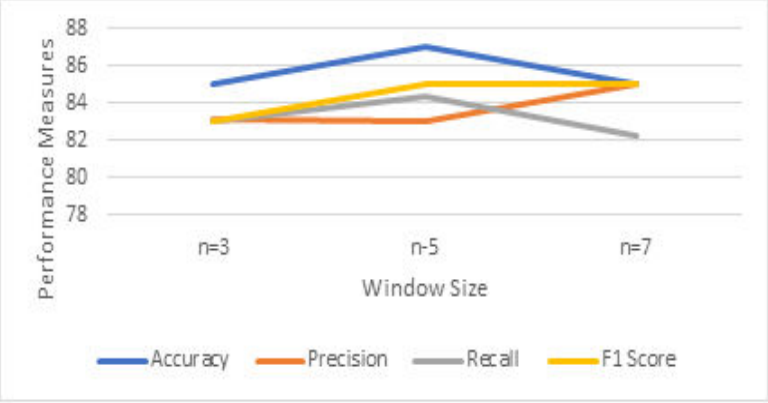


FIGURE 5.4: Performance of CBOW with SVM.

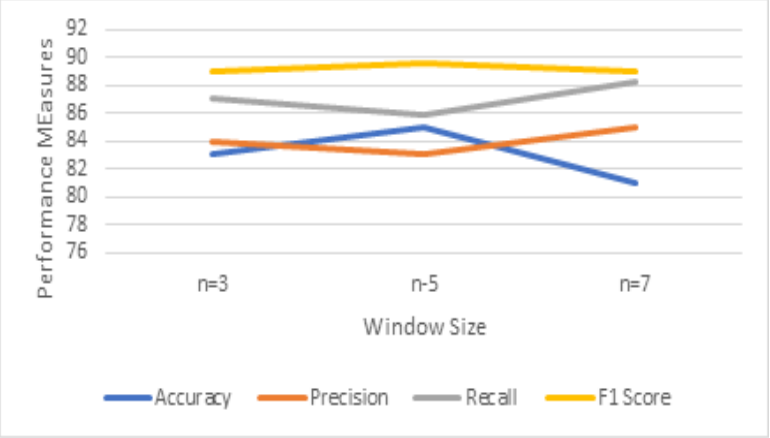


FIGURE 5.5: Performance of Skip-gram with Decision Tree.

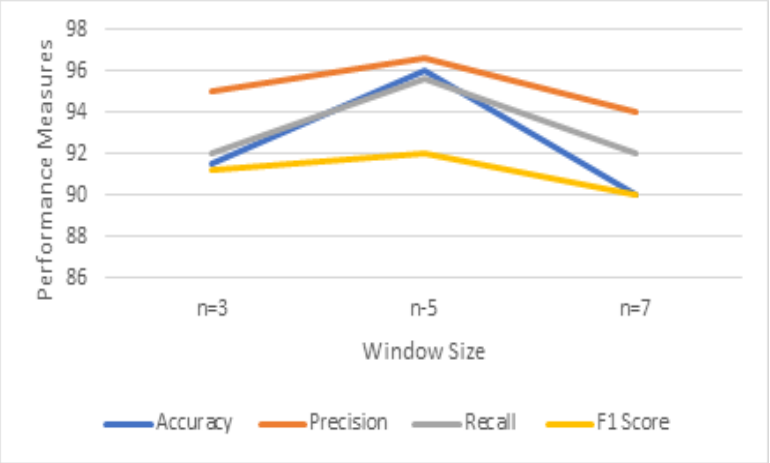


FIGURE 5.6: Performance of Skip-gram with Random Forest.

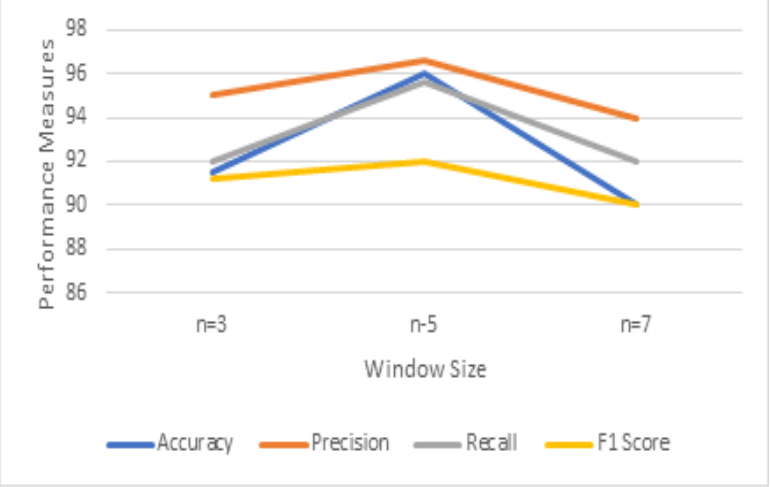


FIGURE 5.7: Performance of Skip-gram with SVM.

5.2.3 Results for Technical Features

The outcomes of several models based on attack patterns retrieved from unstructured CTI reports are assessed in this section. Accuracy, Precision, Recall, and F1-measure are the performance metrics employed in this work. The accuracy of model tells us about its overall performance. Precision indicates the percentage of tuples that the classifier labels as positive. Recall, also called sensitivity, indicates the percentage of relevant results correctly predicted by the classifier. The harmonic mean of Precision and Recall is the F1 measure. These metrics were calculated using the formulas described below. The training-to-testing data ratio is 80:20. The formulas for these metrics are defined below.

Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision = TP / (TP + FP)

Recall = TP/ (TP + FN)

F1-Measure = 2 x ((P x R) / (P+R))

Figure 5.8 depicts the implementation of a heat map. It is an effective way to examine and comprehend complex data relationships as they are commonly used in data analysis and machine learning to generate insights. It is a data visualization that employs color to convey the magnitude of phenomena or the strength of a link between elements. It is especially effective in displaying relationships and trends in vast datasets. These show the relationship between several features in a dataset. Each cell in the matrix indicates the correlation between two variables, with the color intensity or shading representing the degree and direction (positive or negative) of the connection.

True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) are computed for each class separately in a multi-class classification task. TP is the number of correctly predicted instances of a class (e.g., Class 0 in this confusion matrix; the value on the main diagonal is 2145 for Class 0). As the values in the row for that class excluding the diagonal element, such as 12, 0, 13, etc., for Class 0, FN is the total of misclassified instances that truly belong to that class but were expected to be other classes. FP is the number of cases that belong to other classes but were mistakenly anticipated to be that class (values in

the column for that class excluding the diagonal element). TN is the total number of accurately categorized instances (all other values outside that row and column) that are neither projected to belong to that class nor do so in reality. Metrics like precision, recall, and F1-score can then be assessed across all classes by computing TP, FP, FN, and TN for each class.

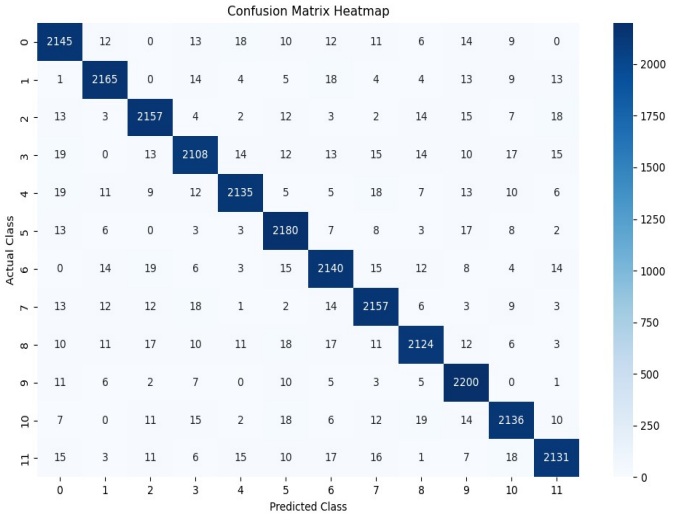


FIGURE 5.8: Heat Map for Technical Features.

Figure 5.9 depicts independent and dependent feature sets in Python implementation. Independent features are the variables that the machine learning model uses as input. They are the predictors or factors on which the model bases its predictions. The qualities, properties, or variables that are changed or controlled to monitor their effect on the dependent variable are referred to as independent features. These variables are utilized to train the model and predict the dependent variable. Dependent features are the variables that the machine learning model attempts to predict using the independent variables. The model attempts to explain or forecast the result or variable of interest. The model is trained on historical data that includes both the independent and dependent variables, and it learns the link between independent and dependent variables.

The purpose of developing a machine learning model is to establish a relationship

between the independent and dependent variables, allowing the model to make accurate predictions on fresh, unseen data based on the patterns learned from the training data. It is critical to select the independent qualities that have a considerable influence on forecasting the dependent variable. To provide the best representation of the relationships between independent and dependent variables, feature selection, engineering, and pre-processing are critical phases in preparing the data for a machine learning model.

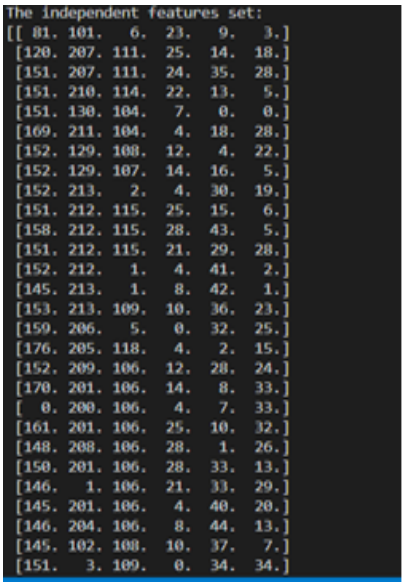


FIGURE 5.9: Independent and Dependent Features.

Figure 5.10 depicts the sample shape of a data collection in Python implementation. Understanding the form of the dataset is critical for various pre-processing activities as well as effectively configuring machine learning models, as the data input shape should match the input layer of the model. Curse of dimensionality can occur when there are many features or dimensions. Understanding the shape aids in procedures such as dimensionality reduction and feature selection. Knowing the shape of the dataset aids in visualizing it, comprehending its structure, and selecting appropriate visualization approaches. Understanding and interpreting dataset shapes is critical for working well with machine learning models. Cross-validation is employed in this work. It splits the original data collection into two halves. The dataset is partitioned into k-sections with k equal to 10. The amount

```
The independent features set:
[[ 81. 101.   6.  23.   9.   3.]
 [120. 207. 111.  25.  14.  18.]
 [151. 207. 111.  24.  35.  28.]
 [151. 210. 114.  22.  13.   5.]
 [151. 130. 104.   7.   0.   0.]
 [169. 211. 104.   4.  18.  28.]
 [152. 129. 108.  12.   4.  22.]
 [152. 129. 107.  14.  16.   5.]
 [152. 213.   2.   4.  30.  19.]
 [151. 212. 115.  25.  15.   6.]
 [158. 212. 115.  28.  43.   5.]
 [151. 212. 115.  21.  29.  28.]
 [152. 212.   1.   4.  41.   2.]
 [145. 213.   1.   8.  42.   1.]
 [153. 213. 109.  10.  36.  23.]
 [159. 206.   5.   0.  32.  25.]
 [176. 205. 118.   4.   2.  15.]
 [152. 209. 106.  12.  28.  24.]
 [170. 201. 106.  14.   8.  33.]
 [  0. 200. 106.   4.   7.  33.]
 [161. 201. 106.  25.  10.  32.]
 [148. 208. 106.  28.   1.  26.]
 [150. 201. 106.  28.  33.  13.]
 [146.   1. 106.  21.  33.  29.]
 [145. 201. 106.   4.  40.  20.]
 [146. 204. 106.   8.  44.  13.]
 [145. 102. 108.  10.  37.   7.]
 [151.   3. 109.   0.  34.  34.]
```

FIGURE 5.10: Shape of Dataset.

and frequency of reports on a cyber-threat actor can vary greatly depending on their activity, visibility, and impact on various organizations or industries. These reports can come from a variety of sources, including cyber-security companies, government agencies, and independent security experts. Table 5.1 shows the number of reports for each CTA.

5.2.4 Individual Threat-Actor Performance

Individual threat actor performance reveals that the majority of threat actors are correctly predicted. APT3, APT17, APT28, APT29, Deep-panda, Fin7, Lazarus, Menu pass, and Oilrig have 100% Precision, Recall, and f1-measure. Rocket kitten has 80% Precision, 100% Recall, and 89% f1-measure, while Turla threat actor has 80% Precision, 80% Recall, and 80% F1-measure, and Winntie threat actor has 100% Precision, 67% Recall, and 80% F1 measure as depicted in table 5.2.

TABLE 5.1: Cyber-Threat Actors.

Group	Aliases	No. of reports
APT3	Gothic panda, pirpi	2230
APT17	Deputy dog, axiom	2250
APT28	Sednit, Fancy bear	2200
APT29	Euroapt, cozy bear	2290
Deep panda	Shell crew, webmasters	2212
Fin7	Fin7	2280
Lazarus	Hidden Cobra, Zinc	2220
Menu Pass	APT10, Hogfish	2250
Oilrig	APT34, Crambus	2240
Rocket Kitten	Shamoon, Magic Hound	2230
	Total	26,910

TABLE 5.2: Individual CTA Results.

Class	Actor	Precision (%)	Recall (%)	F1-measure (%)
0	APT3	100	100	100
1	APT17	100	100	100
2	APT28	100	100	100
3	APT29	100	100	100
4	Deep Panda	100	100	100
5	Fin7	100	100	100
6	Lazarus	100	100	100
7	Menu pass	100	100	100
8	Oilrig	100	100	100
9	Rocket kitten	80	100	89
10	Turla	80	80	80

These results show that using detailed features in attributing CTA improves the overall performance of the model. Results of the detailed feature set are shown in table 5.3. In this study, many classifiers are used to ascribe CTA. Random forest, decision tree, and support vector machine are the classifiers employed. Random forest outperforms the other two classifiers in terms of Accuracy, Precision, Recall, and F1-measure, with values of 96.6%, 96.68%, 95.58%, and 95.75%, respectively.

Table 5.3: Machine Learning Model Performance.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Measure (%)
Random Forest	96.6	96.68	95.58	95.75
Decision Tree	81	84	81	83
SVM	81	84	81	82

5.2.5 Various Model Performance

In this analysis, the performance of attack2vec was compared to other models employed in the research, as shown in Table 5.4. Accuracy, Precision, Recall, and F1-measure are the performance criteria used for comparison. SMOBI, Word2vec, SIMVER, and modified LSI models were employed for comparative analysis. The results show that this model outperforms the other models. In figure 5.11 comparison of SIMVER vs attack2vec is shown. The figure shows that attack2vec outperforms SIMVER and the performance measures of attack2vec is higher than SIMVER embedding model for individual threat actors.

Table 5.4: Performance of Various Models.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Measure (%)
SMOBI	54.4	63.3	50.8	53.4
Word2vec	84.9	90.9	82.3	85.3
SIMVER	86.5	95.4	83.3	87.9
Modified LSI	94	92	89	89
Attack2vec	96	96.6	95.8	95.75

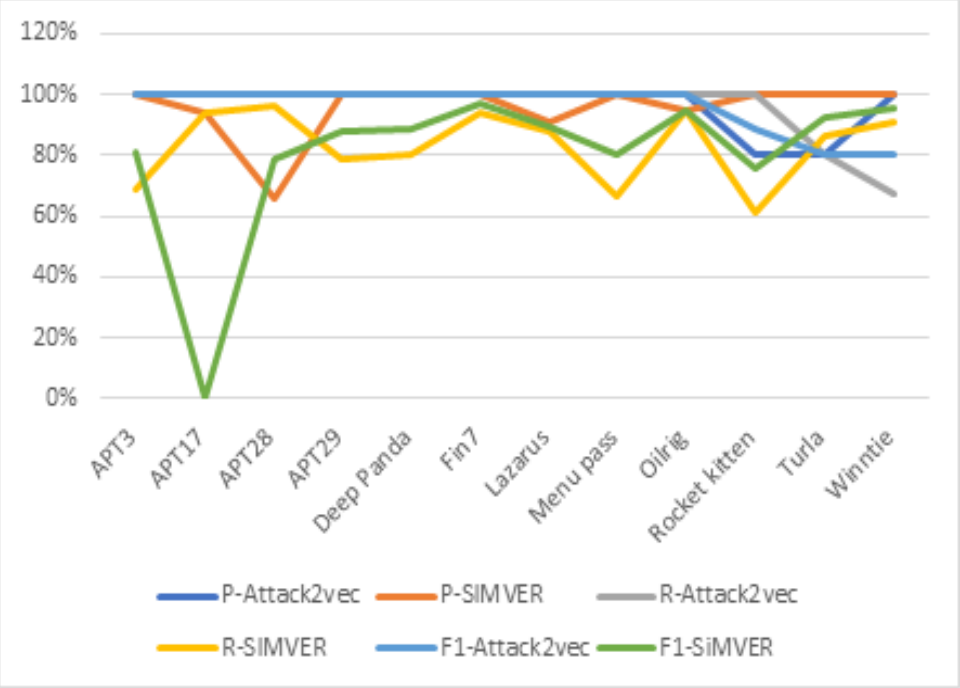


FIGURE 5.11: Individual CTA Results (Attack2vec vs SIMVER.)

5.3 Results for Behavioral Features Attribution

In this section behavioral features attribution results are performed to know the performance of various models. Firstly, the impact of CBOW and skip-gram is evaluated with various machine learning models. The results of behavioral features are discussed in detail.

5.3.1 Model Performance

The performance of the skip-gram and CBOW model is evaluated in this section. Accuracy, Precision, Recall, and F1-measure are the metrics used in this evaluation. Figure 5.12 illustrates the implemented results when CBOW is used with a decision tree. For n=3, results of 81%, 84%, 83, and 81% are obtained. For n=5 results of 84%, 85%, 84%, and 82% are achieved. For n=7 82%, 83%, 83% and 82% is obtained.

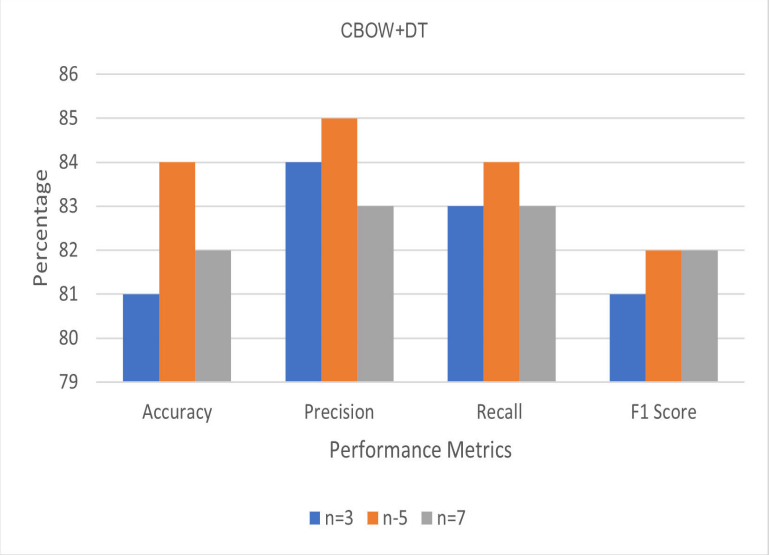


FIGURE 5.12: Performance of CBOW with Decision Tree.

When CBOW is combined with random forest, the implemented results are shown in figure 5.13. Results for n=3 are 81%, 85%, 82%, and 83%. 83%, 88%, 87%, and 87% are the findings obtained for n=5. Results for n=7 are 82%, 85%, 85%, and 84%.

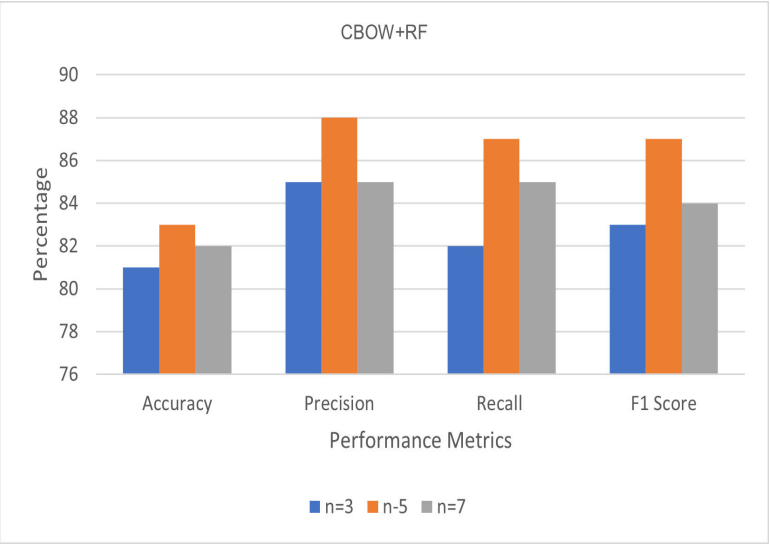


FIGURE 5.13: Performance of CBOW with Random Forest.

Figure 5.14 displays the implemented results of combining CBOW and SVM. For n = 3, the results are 81%, 82%, 81%, and 82%. The results for n=5 are 86%, 83%, 83%, and 86%. The results are 84%, 83%, 82%, and 84% for n=7.

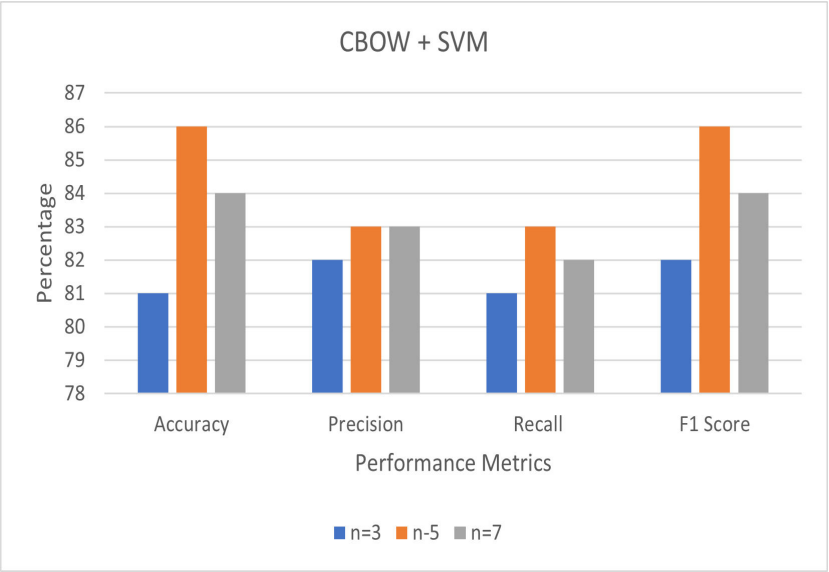


FIGURE 5.14: Performance of CBOW with SVM.

Figure 5.15 displays the implemented results of combining CBOW with LSTM. For n = 3, the results are 91%, 92%, 90%, and 92%. The results for n=5 are 93%, 94%, 93%, and 93%. The results are 91%, 91%, 90%, and 84% for n=7.

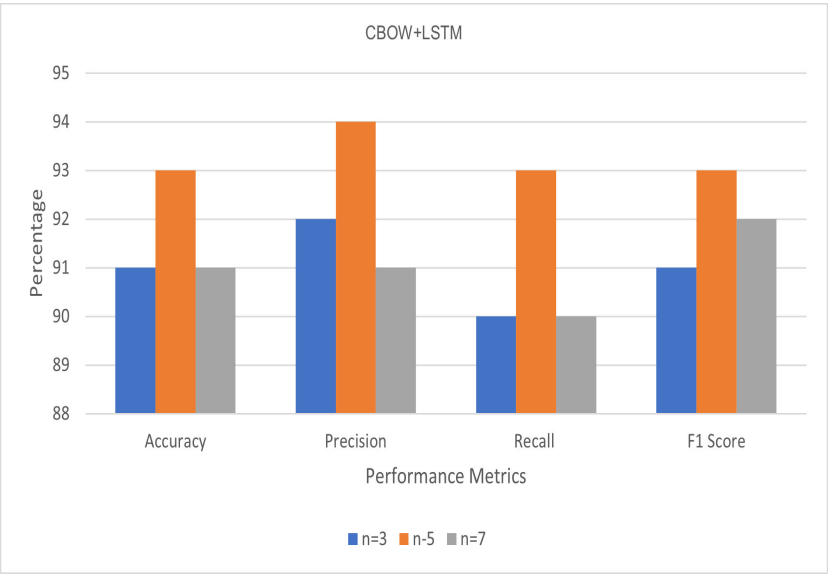


FIGURE 5.15: Performance of CBOW with LSTM.

The results of merging decision trees with skip-gram are shown in figure 5.16. The findings for n = 3 are 81%, 83%, 82%, and 84%. For n = 5, the outcomes are 83%, 85%, 84%, and 85%. For n = 7, the outcomes are 81%, 83%, 83%, and 83%.

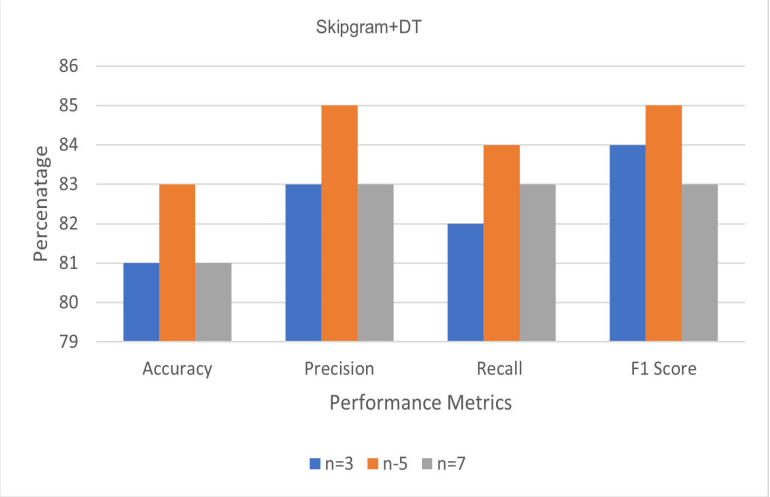


FIGURE 5.16: Performance of Skip-gram with Decision Tree.

Figure 5.17 displays the outcomes of the random forest and skip-gram merger. For n=3, the results are 90%, 94%, 90%, and 90%. The results for n = 5 are 92%, 92%, 94%, and 92%. Results for n = 7 are 92%, 92%, 91%, and 90%.

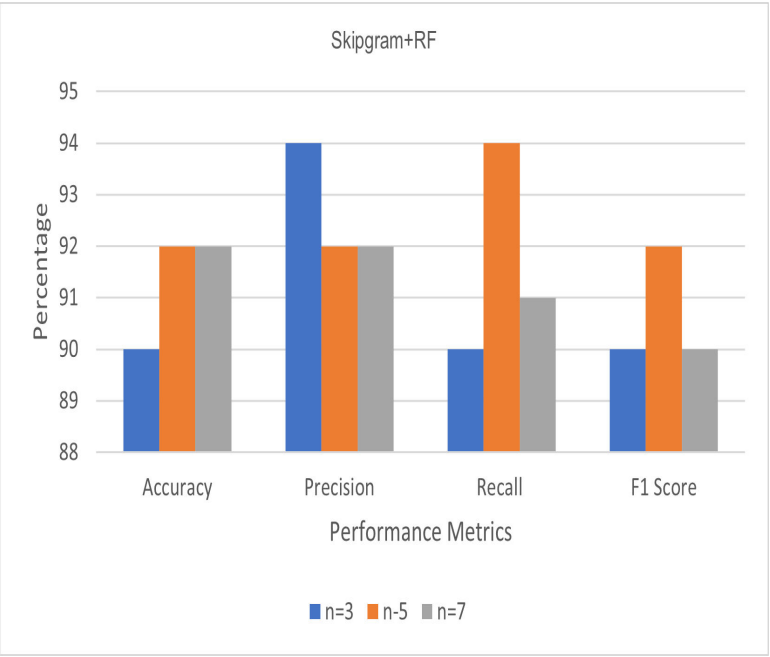


FIGURE 5.17: Performance of Skip-gram with Random Forest.

The results of the merger between SVM and skip-gram are shown in figure 5.18. The findings for n = 3 are 84%, 89%, 84%, and 86%. For n = 5, the findings are 87%, 91%, 90%, and 90%. For n = 7, the findings are 85%, 89.5%, 90%, and 87%.

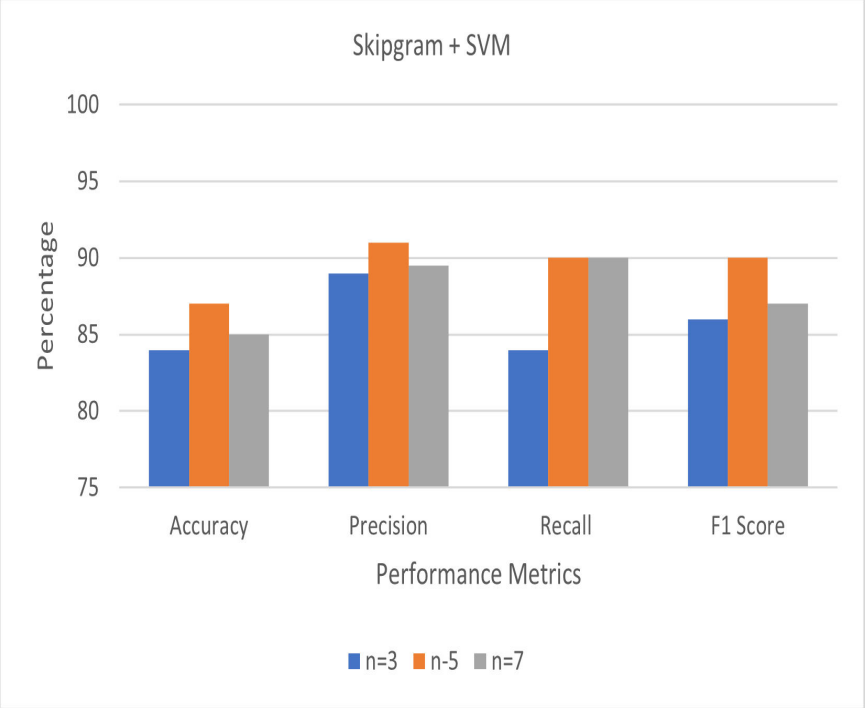


FIGURE 5.18: Performance of Skip-gram with SVM.

The results of merger between LSTM and skip-gram are shown in figure 5.19. The findings for n = 3 are 94%, 93%, 92%, and 90%. For n = 5, findings are 96%, 97%, 94%, and 92%. For n = 7, the findings are 92%, 92%, 91%, and 92%.

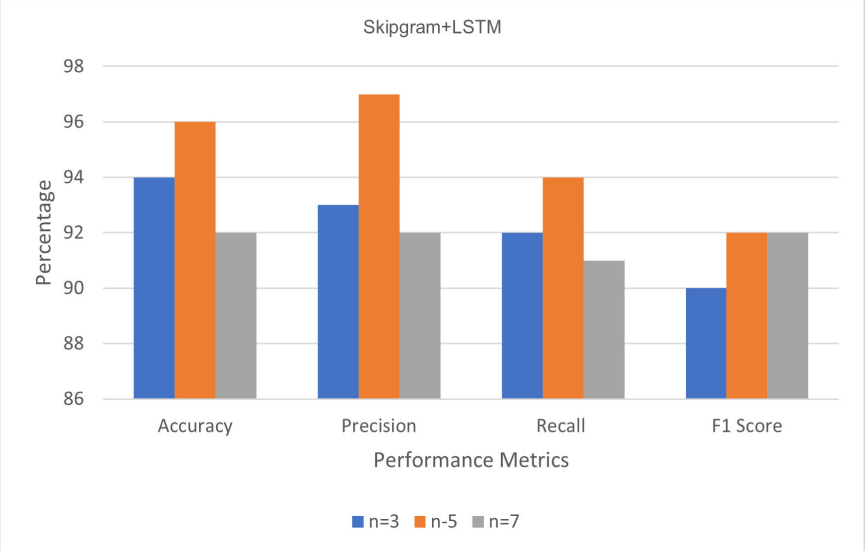


FIGURE 5.19: Performance of Skip-gram with LSTM.

It is clear from the data that the LSTM deep learning algorithm yields the greatest results when paired with skip-gram.

5.3.2 Results for Behavioral Features

In this section, results for behavioral feature extraction are shown. Different features are the subject of experiments and outcomes. First, behavioral aspects are investigated through testing and analysis. As performance parameters, Accuracy, Precision, Recall, and F1-measure are employed. This implementation's test platform is a Core I-7 computer with 16 GB of RAM. Implementation is performed using Python. The IDE is Anaconda.

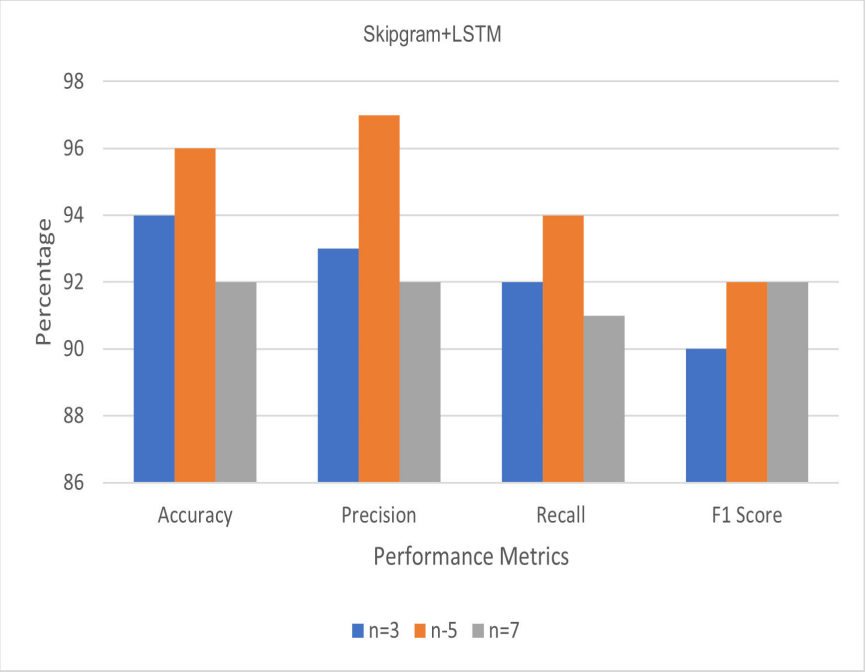


FIGURE 5.20: Performance of Skip-gram with LSTM.

5.3.3 Confusion Matrix

In figure 5.20 confusion matrix is displayed. It is a table that is frequently used in ML to evaluate the performance of a classification system. It is a matrix that displays the model's true-positive, true-negative, false-positive, and false-negative predictions in comparison to the actual ground truth. It is a useful tool for determining where a model thrives and where it falters in its predictions, providing insights into its strengths and shortcomings for various classes or categories being forecast.

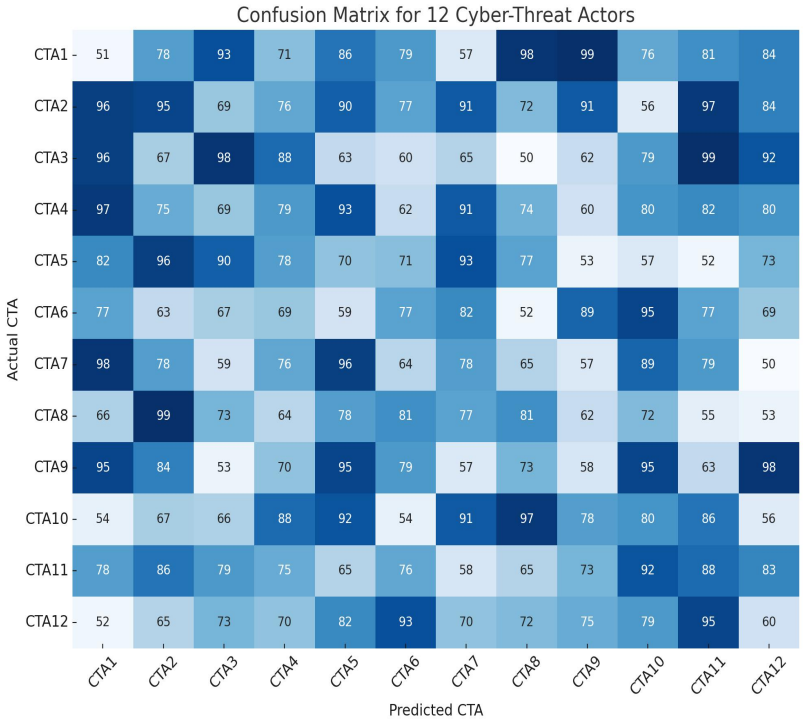


FIGURE 5.21: Confusion Matrix for Behavioral Features.

It is a multi-class classification (twelve cyber-threat actors are used in it). It is a sort of machine learning issue in which the goal is to categorize input data into three or more classes. In this case, the algorithm must assign the input to one of several classes. Each data point is assigned to a single class, and the model is trained to predict the proper class from unseen data. For multi-class classification, common algorithms include logistic regression, decision trees, random forests, support vector machines (SVM), k-nearest neighbors (K-NN), and deep learning approaches such as neural networks. In it, metrics such as Accuracy, Precision, Recall, F1-measure, and confusion matrices can be used to analyze how well the model performs across all classes.

5.3.4 Shape of the Dataset

The geometry of the dataset in the Python implementation is seen in figure 5.21. The independent and dependent feature set is depicted in figure 5.22 For classification, k=10 and k-fold cross-validation are utilized.

```
Shape of the dataset: (248, 7)
     Motivation  Operation Performed  Sponsor By  Origin country  Outcome  Attacker Skills  CTA
0             3                   57         102               6        9                1    0
1             1                  120         207             111       14               18    4
2             1                  151         207             111       35               28    4
3             1                  151         210             114       13                5    4
4             1                  151         130             104        0                0    4
..          ...                  ...         ...             ...      ...              ...  ...
243           1                  106         159               1       37                2   12
244           1                   93         160               1       34               23   12
245           1                  132         161             109       25               25   12
246           1                  133         162               5       20               15   12
247           1                  154         163             118       38               24   12

[248 rows x 7 columns]
```

FIGURE 5.22: Shape of Dataset for Behavioral Features.

```
[248 rows x 7 columns]
0    0
1    1
2    1
3    1
4    1
Name: CTA, dtype: int64
Int64Index([0, 4, 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12], dtype='int64')
The independent features set:
[[  3.  57. 102.   6.   9.]
 [  1. 120. 207. 111.  14.]
 [  1. 151. 207. 111.  35.]
 [  1. 151. 210. 114.  13.]
 [  1. 151. 130. 104.   0.]
 [  1. 169. 211. 104.  18.]]
The dependent variable:
[0 1 1 1 1 1]
```

FIGURE 5.23: Independent and Dependent Variables.

The highest Accuracy, Precision, Recall, and F1-measure for behavioral aspects are recorded at 96%, 97%, 94%, and 92%, respectively, as shown in table 5.5.

TABLE 5.5: Results of Behavioral Features.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Measure (%)
Decision Tree	83	85	84	85
Random Forest	92	94	92	92
SVM	87	91	90	90
LSTM	96	97	94	92

5.4 Results for Hybrid Features Attribution

In this section results for hybrid feature attribution are conducted. Firstly CBOW and Skip-gram model performance is evaluated with various machine learning models.

5.4.1 Model Performance

Skip-gram and CBOW are employed to determine how well various embedding models function. The most recent models in this field are these two. The two models have been put to the test using a variety of classifiers, including decision trees, random forests, support vector machines, and the deep learning classifier long short-term memory (LSTM). Accuracy, Precision, Recall, and F1-measure are the performance metrics used.

Figure 5.23 illustrates the implemented results when CBOW is used with a decision tree. For n=3, results of 82%, 83%, 83, and 82% are obtained. For n=5 results of 84%, 85.8%, 84%, and 84% are achieved . For n=7 results of 81%, 83%, 83% and 82% are obtained. Figure 5.24 illustrates the implemented results when CBOW is used with random forest. For n=3, results of 80%, 83%, 84, and 82% are obtained.

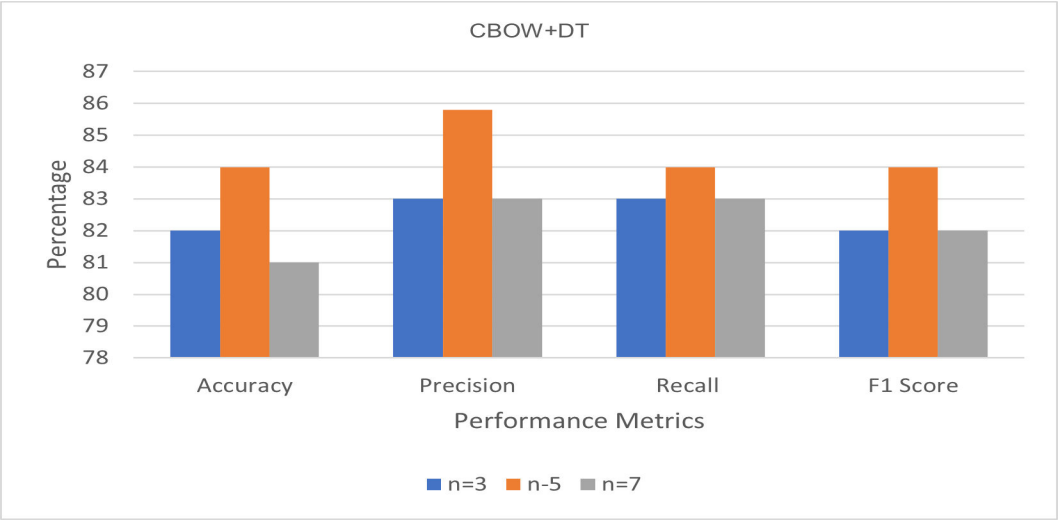


FIGURE 5.24: Performance of CBOW with Decision Tree.

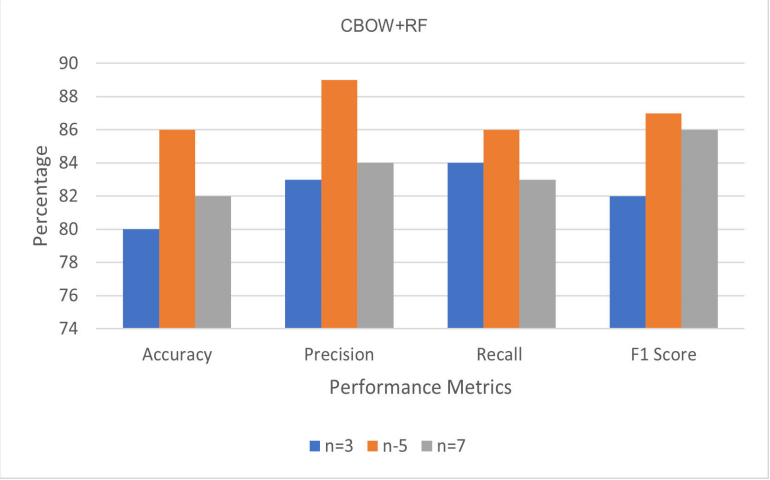


FIGURE 5.25: Performance of CBOW with Random Forest.

For n=5 results of 86%, 89%, 86%, and 87% are achieved. For n=7 results of 82%, 84%, 83% and 86% are obtained. Figure 5.25 illustrates the implemented results when CBOW is used with SVM. For n=3, results of 84%, 82%, 82, and 85% are obtained. For n=5 results of 85%, 84%, 84%, and 87% are achieved. For n=7 results of 84%, 82%, 82% and 84% are obtained. Figure 5.26 illustrates the implemented results when CBOW is used with LSTM. For n=3, results of 90%, 93%, 91%, and 92% are obtained. For n=5 results of 92%, 95%, 94%, and 94% are achieved. For n=7 results of 93%, 93%, 92% and 93% are obtained. Figure 5.27 illustrates the implemented results when skip-gram is used with decision tree. For n=3, results of 83%, 81%, 84, and 82% are obtained. For n=5 results of 85%, 84%, 85%, and 85% are achieved. For n=7 results of 84%, 82%, 84% and 83% are

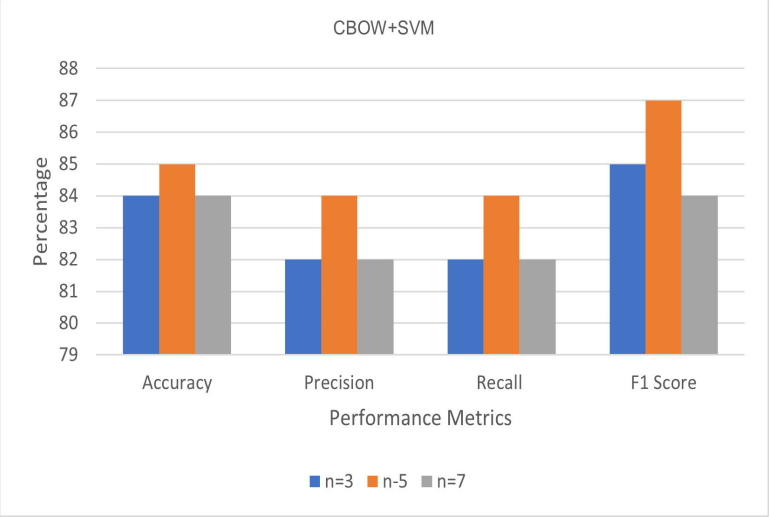


FIGURE 5.26: Performance of CBOW with SVM.

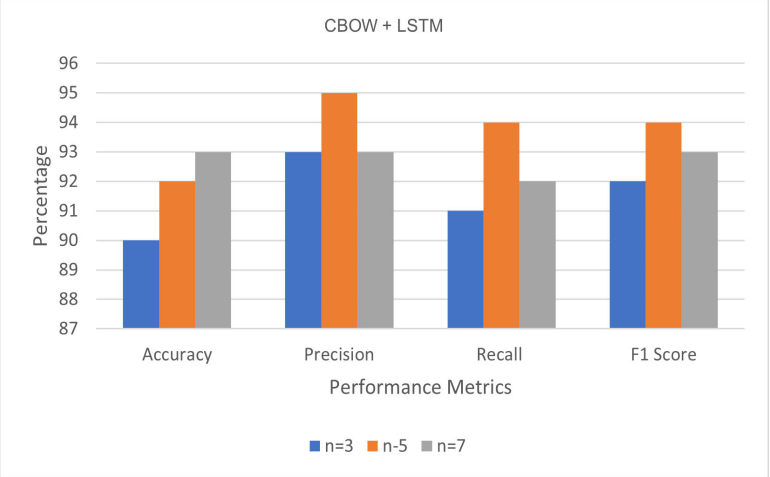


FIGURE 5.27: Performance of CBOW with LSTM.

obtained. Figure 5.28 illustrates the implemented results when skip-gram is used with random forest. For n=3, results of 89%, 93%, 89%, and 91% are obtained. For n=5 results of 91%, 94%, 93%, and 92% are achieved. For n=7 results of 89%, 91%, 92% and 90% are obtained. Figure 5.29 illustrates the implemented results when skip-gram is used with SVM. For n=3, results of 83%, 88%, 84, and 84% are obtained. For n=5 results of 86%, 92%, 91%, and 91% are achieved. For n=7 results of 85%, 90%, 90.5% and 89% are obtained.

Figure 5.30 illustrates the implemented results when skip-gram is used with LSTM. For n=3, results of 94%, 93%, 92, and 90% are obtained. For n=5 results of 97%, 98.5%, 96%, and 96% are achieved. For n=7 results of 93%, 95%, 96% and 95% are obtained. From the experimentation results it is clear that when

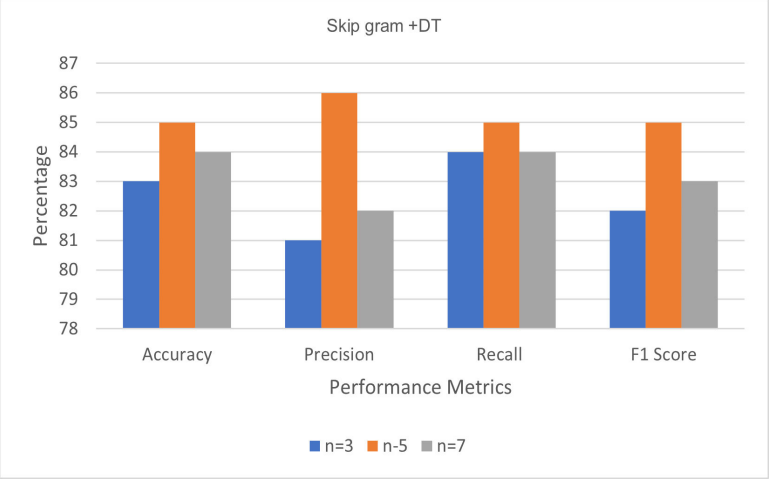


FIGURE 5.28: Skip-gram with Decision Tree.

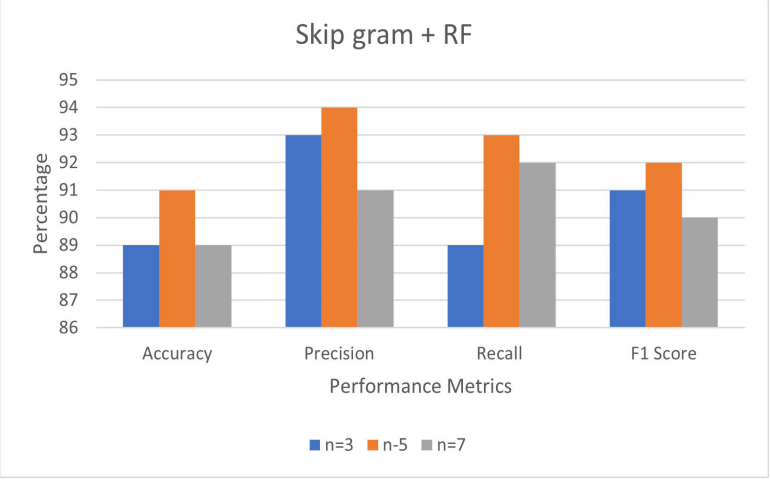


FIGURE 5.29: Skip-gram with Random Forest.

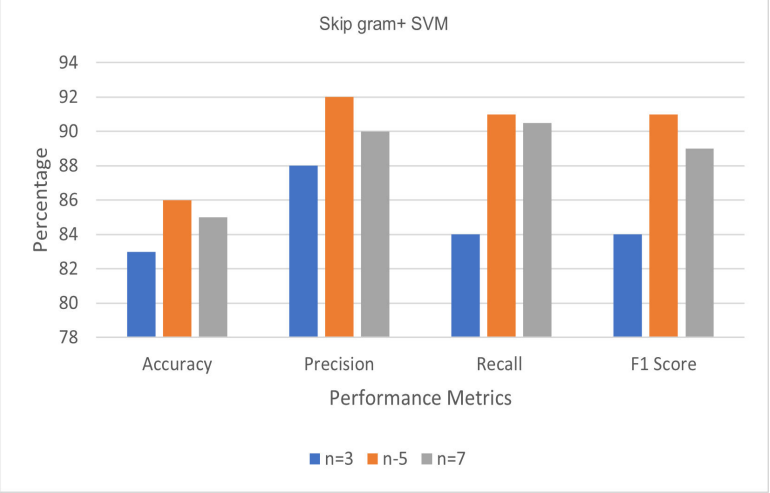


FIGURE 5.30: Skip-gram with SVM

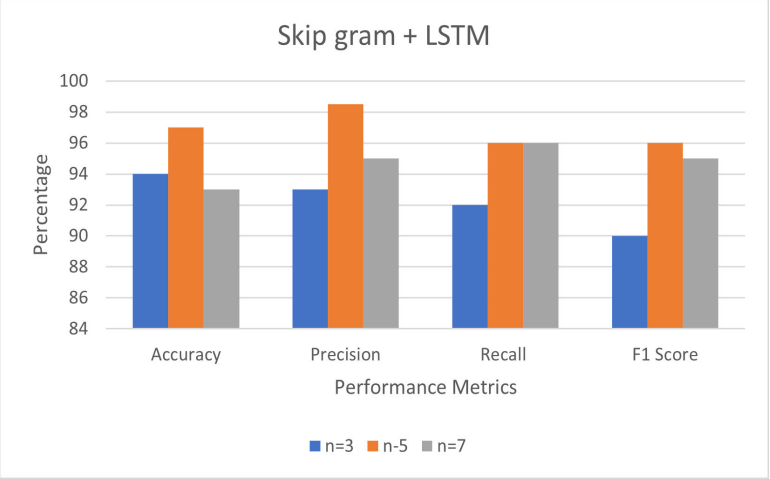


FIGURE 5.31: Skip-gram with LSTM.

combined with skip-gram, the LSTM deep learning algorithm produces the best results.

5.4.2 Results for Hybrid Features

In this section results for hybrid feature selection are shown. Different machine and deep learning algorithms such as decision trees, random forests, and LSTM are used to obtain results. Hybrid features are the subject of experiments and outcomes. Performance parameters, Accuracy, Precision, recall, and F1-measure are employed. This implementation's test platform is a Core I-7 computer with 16 GB of RAM. Implementation is performed using Python. The IDE is Anaconda. For classification, k=10 and k-fold cross-validation are utilized. The influence of hybrid features—that is, features that combine technological and behavioral aspects—is then examined. As indicated in table 5.6, Accuracy, Precision, Recall, and F1-measure were 97%, 98.5%, 96%, and 96% respectively.

5.5 Results for Selected Features

In this section results for optimal feature selection are shown. Different machine and deep learning algorithms such as decision tree, random forest, and LSTM

are used to obtain results. Different machine and deep learning algorithms such as decision tree, random forest, and LSTM are used to obtain results. Hybrid features are the subject of experiments and outcomes.

TABLE 5.6: Results of Hybrid Features.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-measure (%)
Decision Tree	85	86	85	85
Random Forest	91	94	93	92
SVM	86	92	91	91
LSTM	97	98.5	96	96

Performance parameters,Accuracy, Precision, Recall, and F1-measure are employed. This implementation's test platform is a Core I-5 computer with 16 GB of RAM. Implementation is performed using Python. The IDE is Anaconda. For classification, k=10 and k-fold cross-validation are utilized. The Accuracy, Precision, Recall, and F1 measure for optimal features are recorded at 97%, 98.8%, 97%, and 97.2%, respectively, as shown in table 5.7.

TABLE 5.7: Results of Optimal Features

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Measure (%)
Decision Tree	90	92	91	91
Random Forest	91.5	93.8	92	92.4
LSTM	97	98.8	97	97.2

5.6 Accurate Attack Detection in IDS

Several performance evaluation metrics, including Recall, Accuracy, and Precision are employed for experimentation.

5.6.1 NSL-KDD Dataset Results

In this section results for the NSL-KDD dataset are shown. It is a commonly used benchmark dataset for assessing the effectiveness of various machine-learning models for detecting network assaults and evaluating IDS. A full analysis of the results for attack detection would include these measures to provide a holistic view of the IDS performance. Different machine learning methods, such as random forest, and SVM are typically assessed to determine their efficacy in identifying various forms of attacks.

The confusion matrix is shown in figure 5.31. It tells the predicted and actual values. The values that are predicted true by the model against values that are predicted false.

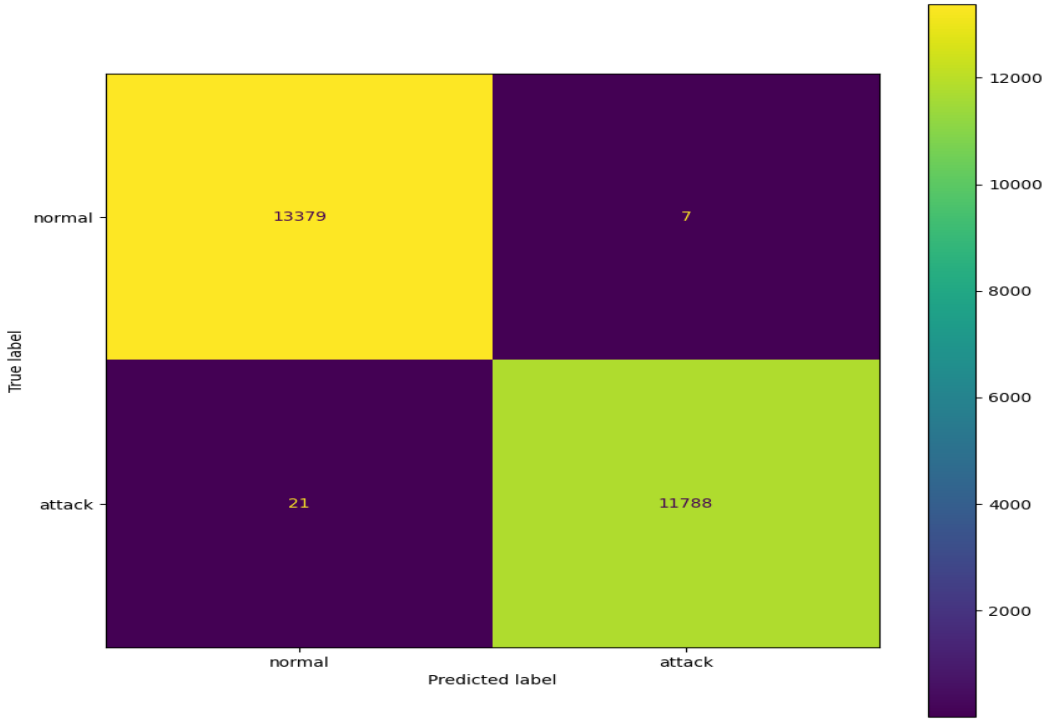


FIGURE 5.32: Confusion Matrix for NSL-KDD Dataset.

Making use of the NSL-KDD data set, the proposed methodology has a 95% Accuracy rate, which is higher than that of existing methods. Using random forest, we achieve Accuracy, Precision, and Recall of 96%, 94%, and 94%, respectively. SVM achieves 94%, 92%, and 94% Accuracy, Precision, and Recall, respectively. The decision tree achieves 92%, 92%, and 91% Accuracy, Precision, and Recall, respectively as shown in table 5.8.

In the experimentation cross validation is used and the value of k=10. The size of the training and testing data set is in the ratio of 80:20. Python is used for implementation. Anaconda is used as an IDE. The test bed for implementation is Core-I-7 processor with 16 GB of RAM. Figure 5.32 shows the results on this dataset.

TABLE 5.8: NSL-KDD Dataset Results

Algorithm	Accuracy (%)	Precision (%)	Recall (%)
Random Forest	96	94	94
SVM	94	92	94
Decision Tree	92	92	91

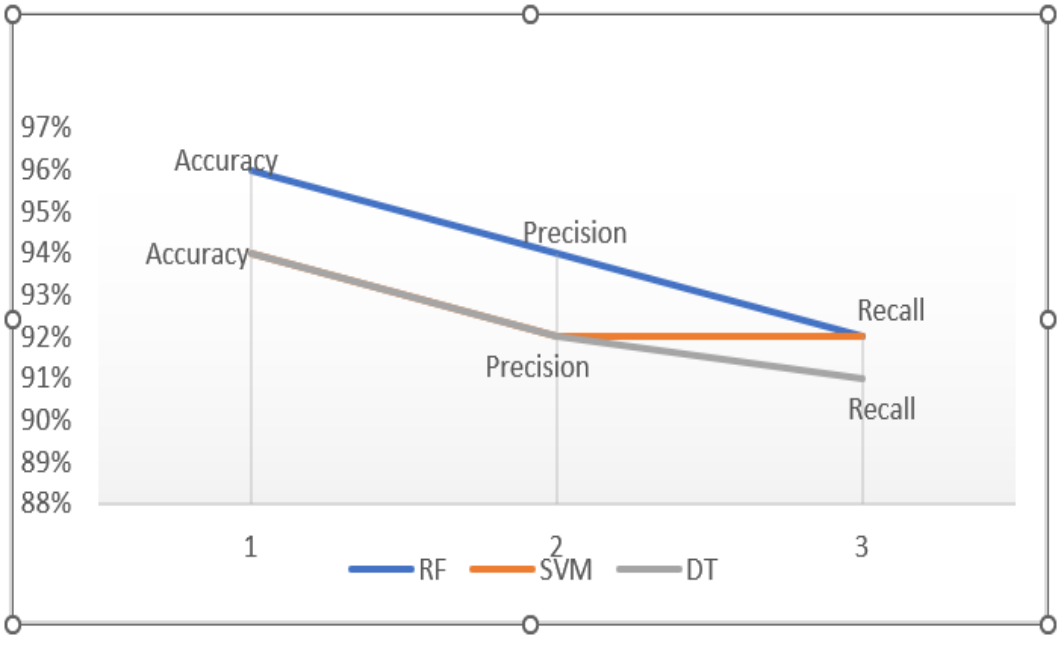


FIGURE 5.33: NSL-KDD Dataset Results.

5.6.2 CSE-CIC-IDS2018 Dataset Results

The proposed methodology achieves an Accuracy of 98% when using the CSE-CIC-IDS2018 dataset, which is greater than that of existing methods. Using random forest, we get 98% Accuracy, 97% Precision, and 96% Recall. SVM produces Accuracy, Precision, and Recall of 94%, 95%, and 95%, respectively. The decision tree achieves 93%, 94%, and 94% Accuracy, Precision, and Recall respectively as shown in table 5.9. Figure 5.33 shows the results comparison.

TABLE 5.9: CIC-IDS2018 Dataset Results

Algorithm	Accuracy (%)	Precision (%)	Recall (%)
Random Forest	98	97	96
SVM	94	95	94
Decision Tree	93	94	94

FIGURE 5.34: CIC-IDS2018 Dataset Results.

5.7 Discussion of Results

1. The performance of skip-gram model is best when used with LSTM for attributing technical features.
2. Random Forest produces high results of 96.6%, 96.8%,95.58%, and 95.75% as compared to decision tree and SVM while attributing technical features.
3. Attack2vec outperforms other models.
4. The performance of the skip-gram model is best when used with LSTM for attributing behavioral features.
5. For attributing hybrid features LSTM produces high results in terms of Accuracy, Precision, Recall, and F1-measure of 96%, 97%, 94%, and 92%.
6. For optimal features the best results are with LSTM having Accuracy, Precision, Recall, and F1-measure of 97%, 98.8%, 97%, and 97.2%.
7. For accurate attack detection using the NSL-KDD dataset random forest outperforms decision tree and SVM having Accuracy, Precision and Recall of 96%, 94%, and 94%.
8. For accurate attack detection using CIC-IDS2018 dataset random forest outperforms decision tree and SVM having Accuracy, Precision, and Recall of 98%, 97%, and 96%.

5.8 Rationale for Using Datasets

The use of these datasets is motivated by the requirement for a comprehensive and high-fidelity approach to cyber-attack attribution process. By using unstructured CTI reports from prominent security vendors like as Fire-eye, Crowd-strike, and Symantec, one gain access to a massive repository of real-world threat data, including a wide range of indicators. Also these reports are used by various researchers in their experimentation. They provide a solid platform for modeling and attributing CTAs. Furthermore, integration of the threat actor encyclopedia dataset allows for the extraction of detailed CTA specific information for behavioral patterns.

The variety of datasets enables the synthesis of both technical and behavioral features, which is critical for developing a hybrid threat attribution paradigm. The size and diversity of the data covered a wide range of threat scenarios, improving the accuracy, scalability of attribution models in the dynamic world of cyber threat intelligence.

The NSK-KDD and CSE-CIC-IDS2018 datasets are most widely used in accurate attack detection in IDS. The NSK-KDD dataset, provides a well-structured and well researched intrusion detection benchmark, making it perfect for testing and evaluating the performance of new IDS models. Its diverse set of attack scenarios and network traffic data serves as a thorough test bed for determining the effectiveness of detection algorithms.

The CSE-CIC-IDS2018 dataset, on the other hand, is more modern and realistic, representing a wide range of contemporary attack patterns in a simulated business setting. It covers current attack vectors and benign traffic, demonstrating the changing nature of cyber threats. By utilizing these datasets, a comprehensive assessment of IDS performance across multiple time periods and threat scenarios is analyzed.

5.9 Limitation of Research Work

The fundamental goal of this research is to extract attack patterns for cyber-threat actors. This investigation identifies CTA more accurately. The experiments confirmed our hypothesis, and the results were correct. Before the proposed approach can be used in a real-world security context, some restrictions must be overcome. There is a dataset limitation in this domain; finding an appropriate one for experimentation is a difficult task. Attack patterns and CTA are frequently documented in the dataset as human-readable enumerations, which are frequently long comprehensive statements that make extracting important information challenging.

Furthermore, many behavioral traits can be used for experimentation but cannot be extracted due to a lack of adequate datasets.

There is no standard framework for reporting (unstructured reports), so extracting valuable information is challenging. As a result, there are skewed datasets. There is no benchmark dataset in this field. Because separate research is done on distinct data, comparing different methodologies is difficult. It is challenging to extract all aspects from a report; missing values are possible.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this research dissertation, conventional methodologies for detecting cyber-threat actors with technical and behavioral features were implemented. The changing nature of the cyber-threat landscape needs a thorough examination of the motivations and backgrounds of CTA. This study investigates how behavioral features and the optimum feature selection affect the attribution of CTA. It is a novel notion in this field. Machine/deep learning models are used in this study to examine the impact of behavioral features. It will help to understand the attacker's background and motivations. The impact of behavioral features is also studied. The optimal feature for CTA attribution is chosen.

There is an enormous amount of CTI information available throughout the world. Threat feeds, hacker forums, social media, the dark web, security websites, threat warnings, honeypots, CVE, NVD, and unstructured CTI reports are some of the sources. Manually extracting relevant information from this data is a difficult task. CTA attribution was conducted in this study by extracting features from unstructured CTI reports. TTP, tools, malware, target organization, country, and application have all been used in detail. This extensive feature set offered a full overview of the attacker profile and aided in more accurate and exact attribution.

The "Attack2vec" model, which is trained on domain-specific embedding, has been developed. The results demonstrated that this innovative model outperforms other models. This unique model achieves 96.5% Accuracy, 96.50% Precision, 95.58% Recall, and 95.75% F1-measure. For categorization, machine learning methods such as decision trees, random forests, and support vector machines have been utilized. The use of a detailed feature set improves classification outcomes.

This research work investigates the impact of behavioral characteristics on cyber-threat attribution. The incorporation of these elements is a unique notion in this field that has yet to be extracted for CTA attribution. The attacker will be more clearly detailed and contextualized by giving behavioral characteristics. Following the completion of tests and the analysis, it is required to evaluate the influence of hybrid features by analyzing their impact on the cyber-threat attribution process. The results of experimentation show that adding behavioral aspects results in good outcomes.

This study has made substantial progress towards tackling critical issues in cyber threat actor attribution by utilizing unstructured CTI reports and threat actor encyclopedia dataset. Explored the extraction of comprehensive technical elements from unstructured CTI reports and assessed the impact of integrating behavioral features from the threat actor encyclopedia. Demonstrated a more complete and successful methodology for attribution of cyber threats by providing a novel hybrid approach that incorporates both technical and behavioral elements.

One of the key research gaps discovered was the limited number of available datasets and their inherent imbalance, which caused issues in feature extraction and attribution accuracy. Our approach successfully mitigates these difficulties by applying advanced feature reduction techniques to the feature set, hence improving the accuracy and reliability of cyber threat attribution.This work not only fills scientific gaps, but also establishes a solid foundation for future research in this area. Identifying CTA is a complex task, using hybrid features will help to identify cyber threat actors more precisely and accurately. Using feature selection/reduction methodologies, the ideal feature set for cyber-threat attribution is then discovered. In comparison to previous models, attained Accuracy, Precision,

Recall, and F1-measure of 97.8%, 98.8%, 97.2%, and 97.2%, respectively, which are good results in this area.

In conclusion, this study improves the field by providing a more refined approach to cyber threat attribution, emphasizing the relevance of combining technical and behavioral aspects, and overcoming dataset restrictions. The suggested model is a big step forward in enhancing the precision of cyber threat intelligence, bringing useful insights and methodologies to the security community.

Rate of cybercrime is rapidly increasing, which is a significant drawback of technology. There are numerous approaches and methods for attackers to breach systems. Researchers developed several solutions based on machine learning algorithms to protect systems from such attackers, which are crucial in detecting and safeguarding assets from a variety of threats. This research investigation proposed a way for more precisely detecting attacks in IDS using machine learning approaches. For testing, the suggested approach employs two well-used data sets: NSL-KDD and CSE-CIC-IDS2018. This methodology yields an overall Accuracy of 96% with the NSL-KDD data set and an Accuracy of 98% with the CSE-CIC-IDS2018 dataset. This proposed method detects network attacks more accurately and precisely than earlier methods.

6.2 Future Work

Future research in this field should focus on identifying a fully automated mechanism for drawing a complete picture of attack flow. There exists in the literature a semi-automated mechanism that draws attack flow. But so far in the research, no fully automated mechanism exists. It will help in better identifying CTA.

Identifying trends and patterns in the attribution of CTA is an important task. The grouping of CTAs according to attack patterns will aid in connecting related CTA based on the attack models they employ. It will aid in designing a thorough offensive strategy. Deep learning techniques will be employed in the future to improve classification results for better and more accurate IDS attack detection. Future studies on this topic will concentrate on discovering trends and patterns

in the attribution of cyber-threat actors based on numerous variables. Threat intelligence relies on technical and behavioral characteristics that explain an adversary's behavior and attack patterns. Identifying regularities among them will improve the process of attribution of cyber threats. Early assault detection can result in an efficient technique for evaluating CTAs and their attack pathways. If CTAs are classified according to attack patterns, it will be easier to connect relevant cyber-threat actors based on the attack models they utilize. It will aid in the development of a comprehensive attack strategy.

Deeper understanding of cyber-threat actor attribution may be possible in future research if malware types (virus, Trojan,ransomware,spyware) are incorporated. This can improve the attribution models capacity to differentiate between threat actors according to their preferred malware tools by classifying and examining malware families, variations, and the behaviors that go along with them. Given that some threat actors tend to depend on specific malware type, future research would enable the identification of distinctive attack patterns more accurately.

Future research in this domain could focus on improving attribution accuracy and granularity by combining advanced machine learning algorithms with larger and more diversified datasets. This includes investigating real-time attribution systems that use both technical indications (such as network traffic and malware signatures) and behavioral characteristics (such as attack patterns and sociopolitical circumstances). Another interesting avenue is the creation of standardized frameworks for sharing and correlating threat intelligence across organizations and industries, which could boost coordinated efforts to combat cyber threats. Furthermore, tackling the issues of attribution in new technologies such as IoT, cloud computing, and quantum computing is critical. Finally, in order to strike a balance between security and civil freedoms, ethical considerations such as privacy and preventing the misuse of attribution data should be prioritized.

Large Language Models (LLMs) serve an important role in enhancing the cyber-attack attribution process by overcoming the challenges of analyzing unstructured threat data. Traditional approaches frequently struggle to handle the vast volume and complexity of data created in cyber threat landscapes, which comprise a wide

range of sources such as academic publications, vendor reports, and open-source intelligence. LLMs excel at natural language understanding and generation, allowing them to process and extract useful information from large amounts of textual data. They can detect trends, correlations, and subtle contextual indicators that conventional algorithms may miss, improving the accuracy of threat actor profiling. Furthermore, LLMs can adapt to changing threat landscapes by constantly learning from fresh data, ensuring that the attribution process remains current and adaptable to emerging threats. Their ability to integrate technical details with behavioral research contributes to the hybrid threat attribution technique, which provides a holistic perspective of cyber threats. This not only increases the speed and accuracy of attribution, but it also helps to forecast future attacks by understanding threat actors motivations and plans, resulting in more effective cyber-security defenses and proactive threat mitigation.

Bibliography

[1] T. Hashem, "Examining marketing cyber-security in the digital age: Evidence from marketing platform," *International Journal of Data and Network Science*, vol. 8, no. 2, pp. 1141–1150, 2024.

[2] M. Haugli-Sandvik, M. S. Lund, and F. B. Bjørneseth, "Maritime decision-makers and cyber security: deck officers perception of cyber risks towards it and ot systems," *International Journal of Information Security*, pp. 1–19, 2024.

[3] A. J. G. de Azambuja, T. Giese, K. Schützer, R. Anderl, B. Schleich, and V. R. Almeida, "Digital twins in industry 4.0–opportunities and challenges related to cyber security," *Procedia CIRP*, vol. 121, pp. 25–30, 2024.

[4] A. Goni, M. U. F. Jahangir, and R. R. Chowdhury, "A study on cyber security: Analyzing current threats, navigating complexities, and implementing prevention strategies," *International Journal of Research and Scientific Innovation*, vol. 10, no. 12, pp. 507–522, 2024.

[5] M. Zwilling, G. Klien, D. Lesjak, Ł. Wiechetek, F. Cetin, and H. N. Basim, "Cyber security awareness, knowledge and behavior: A comparative study," *Journal of Computer Information Systems*, vol. 62, no. 1, pp. 82–97, 2022.

[6] K. Rajasekharaiah, C. S. Dule, and E. Sudarshan, "Cyber security challenges and its emerging trends on latest technologies," in *IOP Conference Series: Materials Science and Engineering*, vol. 981, pp. 22–30, IOP Publishing, 2020.

[7] A. M. Tonge, S. S. Kasture, and S. R. Chaudhari, "Cyber security: challenges for society-literature review," *IOSR Journal of computer Engineering*, vol. 2, no. 12, pp. 67–75, 2013.

[8] R. Von Solms and J. Van Niekerk, "From information security to cyber security," *computers & security*, vol. 38, pp. 97–102, 2013.

[9] M. McNeese, N. J. Cooke, A. D'Amico, M. R. Endsley, C. Gonzalez, E. Roth, and E. Salas, "Perspectives on the role of cognition in cyber security," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 56, pp. 268–271, SAGE Publications Sage CA: Los Angeles, CA, 2012.

[10] K.-K. R. Choo, "The cyber threat landscape: Challenges and future research directions," *Computers & security*, vol. 30, no. 8, pp. 719–731, 2011.

[11] A. Spence and S. Bangay, "Security beyond cybersecurity: side-channel attacks against non-cyber systems and their countermeasures," *International Journal of Information Security*, vol. 21, no. 3, pp. 437–453, 2022.

[12] S. Achar, "Cloud computing security for multi-cloud service providers: Controls and techniques in our modern threat landscape," *International Journal of Computer and Systems Engineering*, vol. 16, no. 9, pp. 379–384, 2022.

[13] M. M. Nair, A. Deshmukh, and A. K. Tyagi, "Artificial intelligence for cyber security: Current trends and future challenges," *Automated Secure Computing for Next-Generation Systems*, pp. 83–114, 2024.

[14] A. K. Tyagi, "Blockchain and artificial intelligence for cyber security in the era of internet of things and industrial internet of things applications," in *AI and Blockchain Applications in Industrial Robotics*, pp. 171–199, IGI Global, 2024.

[15] M. F. Rafy, "Artificial intelligence in cyber security," *Available at SSRN 4687831*, 2024.

[16] M. Ozkan-Ozay, E. Akin, Ö. Aslan, S. Kosunalp, T. Iliev, I. Stoyanov, and I. Beloev, "A comprehensive survey: Evaluating the efficiency of artificial

intelligence and machine learning techniques on cyber security solutions," *IEEE Access*, 2024.

[17] M. Aslam, "Ai and cybersecurity: An ever-evolving landscape," *International Journal of Advanced Engineering Technologies and Innovations*, vol. 1, no. 1, pp. 52–71, 2024.

[18] E. Ukwandu, M. A. Ben-Farah, H. Hindy, M. Bures, R. Atkinson, C. Tachtatzis, I. Andonovic, and X. Bellekens, "Cyber-security challenges in aviation industry: A review of current and future trends," *Information*, vol. 13, no. 3, p. 146, 2022.

[19] S. Mahmood, M. Chadhar, and S. Firmin, "Cybersecurity challenges in blockchain technology: A scoping review," *Human Behavior and Emerging Technologies*, vol. 2022, pp. 1–11, 2022.

[20] F. Akpan, G. Bendiab, S. Shiaeles, S. Karamperidis, and M. Michaloliakos, "Cybersecurity challenges in the maritime sector," *Network*, vol. 2, no. 1, pp. 123–138, 2022.

[21] S. Ainslie, D. Thompson, S. Maynard, and A. Ahmad, "Cyber-threat intelligence for security decision-making: A review and research agenda for practice," *Computers & Security*, pp. 103–119, 2023.

[22] T. D. Wagner, K. Mahbub, E. Palomar, and A. E. Abdallah, "Cyber threat intelligence sharing: Survey and research directions," *Computers & Security*, vol. 87, pp. 101–589, 2019.

[23] W. Ge and J. Wang, "Seqmask: behavior extraction over cyber threat intelligence via multi-instance learning," *The Computer Journal*, vol. 67, no. 1, pp. 253–273, 2024.

[24] M. Al-Hawawreh, N. Moustafa, and J. Slay, "A threat intelligence framework for protecting smart satellite-based healthcare networks," *Neural Computing and Applications*, vol. 36, no. 1, pp. 15–35, 2024.

[25] C. Sillaber, C. Sauerwein, A. Mussmann, and R. Breu, "Data quality challenges and future research directions in threat intelligence sharing practice," in *Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security*, pp. 65–70, 2016.

[26] E. W. Burger, M. D. Goodman, P. Kampanakis, and K. A. Zhu, "Taxonomy model for cyber threat intelligence information exchange technologies," in *Proceedings of the 2014 ACM Workshop on Information Sharing & Collaborative Security*, pp. 51–60, 2014.

[27] A. N. Irfan, S. Chuprat, M. N. Mahrin, and A. Ariffin, "Taxonomy of cyber threat intelligence framework," in *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1295–1300, IEEE, 2022.

[28] P. Panagiotou, C. Iliou, K. Apostolou, T. Tsikrika, S. Vrochidis, P. Chatzimisios, and I. Kompatsiaris, "Towards selecting informative content for cyber threat intelligence," in *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, pp. 354–359, IEEE, 2021.

[29] K. Abinesh Kamal and S. Divya, "Integrated threat intelligence platform for security operations in organizations," *Automatika*, vol. 65, no. 2, pp. 401–409, 2024.

[30] M. R. Labu and M. F. Ahammed, "Next-generation cyber threat detection and mitigation strategies: A focus on artificial intelligence and machine learning," *Journal of Computer Science and Technology Studies*, vol. 6, no. 1, pp. 179–188, 2024.

[31] O. Kayode-Ajala, "Applications of cyber threat intelligence (cti) in financial institutions and challenges in its adoption," *Applied Research in Artificial Intelligence and Cloud Computing*, vol. 6, no. 8, pp. 1–21, 2023.

[32] F. Menges, C. Sperl, and G. Pernul, "Unifying cyber threat intelligence," in *Trust, Privacy and Security in Digital Business: 16th International Conference, TrustBus 2019, Linz, Austria, August 26–29, 2019, Proceedings 16*, pp. 161–175, Springer, 2019.

[33] M. Conti, T. Dargahi, and A. Dehghantanha, *Cyber threat intelligence: challenges and opportunities*. Springer, 2018.

[34] R. Brown and R. M. Lee, "The evolution of cyber threat intelligence (cti): 2019 sans cti survey," *SANS Institute. Available online: https://www. sans. org/white-papers/38790/(accessed on 12 July 2021)*, 2019.

[35] F. Sadique, S. Cheung, I. Vakilinia, S. Badsha, and S. Sengupta, "Automated structured threat information expression (stix) document generation with privacy preservation," in *2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pp. 847–853, IEEE, 2018.

[36] F. Marchiori, M. Conti, and N. V. Verde, "Stixnet: A novel and modular solution for extracting all stix objects in cti reports," *arXiv preprint arXiv:2303.09999*, 2023.

[37] D. Safavi-Zadeh, "Unravelling the information asymmetry in threat intelligence," B.S. thesis, University of Twente, 2024.

[38] D. P. Möller, "Threats and threat intelligence," in *Guide to Cybersecurity in Digital Transformation: Trends, Methods, Technologies, Applications and Best Practices*, pp. 71–129, Springer, 2023.

[39] B. M. Ampel, S. Samtani, H. Zhu, H. Chen, and J. F. Nunamaker Jr, "Improving threat mitigation through a cybersecurity risk management framework: A computational design science approach," *Journal of Management Information Systems*, vol. 41, no. 1, pp. 236–265, 2024.

[40] R. Trifonov, O. Nakov, and V. Mladenov, "Artificial intelligence in cyber threats intelligence," in *2018 international conference on intelligent and innovative computing applications (ICONIC)*, pp. 1–4, IEEE, 2018.

[41] J. Kotsias, A. Ahmad, and R. Scheepers, "Adopting and integrating cyber-threat intelligence in a commercial organisation," *European Journal of Information Systems*, vol. 32, no. 1, pp. 35–51, 2023.

[42] K. Nova, "Security and resilience in sustainable smart cities through cyber threat intelligence," *International Journal of Information and Cybersecurity*, vol. 6, no. 1, pp. 21–42, 2022.

[43] D. Schlette, M. Caselli, and G. Pernul, "A comparative study on cyber threat intelligence: The security incident response perspective," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 4, pp. 2525–2556, 2021.

[44] S. Samtani, M. Abate, V. Benjamin, and W. Li, "Cybersecurity as an industry: A cyber threat intelligence perspective," *The Palgrave Handbook of International Cybercrime and Cyberdeviance*, pp. 135–154, 2020.

[45] K. Oosthoek and C. Doerr, "Cyber threat intelligence: A product without a process?," *International Journal of Intelligence and CounterIntelligence*, vol. 34, no. 2, pp. 300–315, 2021.

[46] M. H. U. Sharif and M. A. Mohammed, "A literature review of financial losses statistics for cyber security and future trend," *World Journal of Advanced Research and Reviews*, vol. 15, no. 1, pp. 138–156, 2022.

[47] A. Saravanan and S. S. Bama, "A review on cyber security and the fifth generation cyberattacks," *Oriental journal of computer science and technology*, vol. 12, no. 2, pp. 50–56, 2019.

[48] Z. Zhan, M. Xu, and S. Xu, "Predicting cyber attack rates with extreme values," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 8, pp. 1666–1677, 2015.

[49] B. Bouyeddou, F. Harrou, B. Kadri, and Y. Sun, "Detecting network cyber-attacks using an integrated statistical approach," *Cluster Computing*, vol. 24, pp. 1435–1453, 2021.

[50] A. F. Altwairqi, M. A. AlZain, B. Soh, M. Masud, and J. Al-Amri, "Four most famous cyber attacks for financial gains," *Int. J. Eng. Adv. Technol*, vol. 9, pp. 2131–2139, 2019.

[51] A. Berndt and J. Ophoff, "Exploring the value of a cyber threat intelligence function in an organization," in *Information Security Education. Information Security in Action: 13th IFIP WG 11.8 World Conference, WISE 13, Maribor, Slovenia, September 21–23, 2020, Proceedings 13*, pp. 96–109, Springer, 2020.

[52] K. Dunnett, S. Pal, and Z. Jadidi, "Challenges and opportunities of blockchain for cyber threat intelligence sharing," *Secure and Trusted Cyber Physical Systems: Recent Approaches and Future Directions*, pp. 1–24, 2022.

[53] M. Olaifa, J. J. van Vuuren, D. Du Plessis, and L. Leenen, "Security issues in cyber threat intelligence exchange: A review," in *Science and Information Conference*, pp. 1308–1319, Springer, 2023.

[54] X. Liao, K. Yuan, X. Wang, Z. Li, L. Xing, and R. Beyah, "Acing the ioc game: Toward automatic discovery and analysis of open-source cyber threat intelligence," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 755–766, 2016.

[55] U. Noor, Z. Anwar, J. Altmann, and Z. Rashid, "Customer-oriented ranking of cyber threat intelligence service providers," *Electronic Commerce Research and Applications*, vol. 41, pp. 1567–1587, 2020.

[56] T. Schaberreiter, V. Kupfersberger, K. Rantos, A. Spyros, A. Papanikolaou, C. Ilioudis, and G. Quirchmayr, "A quantitative evaluation of trust in the quality of cyber threat intelligence sources," in *Proceedings of the 14th international conference on availability, reliability and security*, pp. 1–10, 2019.

[57] A. Ramsdale and N. Shiaeles, Stavros, "A comparative analysis of cyber-threat intelligence sources, formats and languages," *Electronics*, vol. 9, no. 5, p. 824, 2020.

[58] R. Al-Shaer, J. M. Spring, and E. Christou, "Learning the associations of mitre att&ck adversarial techniques," in *2020 IEEE Conference on Communications and Network Security (CNS)*, pp. 1–9, IEEE, 2020.

[59] R. Kwon, T. Ashley, J. Castleberry, P. Mckenzie, and S. N. G. Gourisetti, "Cyber threat dictionary using mitre att&ck matrix and nist cybersecurity framework mapping," in *2020 Resilience Week (RWS)*, pp. 106–112, IEEE, 2020.

[60] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "Mitre att&ck (trademark): design and philosophy," *MITRE Corporation, McLean, VA*, 2018.

[61] M. Ahmed, S. Panda, C. Xenakis, and E. Panaousis, "Mitre att&ck driven cyber risk assessment," in *Proceedings of the 17th International Conference on Availability, Reliability and Security*, pp. 1–10, 2022.

[62] A. Yousaf and J. Zhou, "From sinking to saving: Mitre att &ck and d3fend frameworks for maritime cybersecurity," *International Journal of Information Security*, pp. 1–16, 2024.

[63] P. N. Bahrami, A. Dehghantanha, T. Dargahi, R. M. Parizi, K.-K. R. Choo, and H. H. Javadi, "Cyber kill chain-based taxonomy of advanced persistent threat actors: Analogy of tactics, techniques, and procedures," *Journal of information processing systems*, vol. 15, no. 4, pp. 865–889, 2019.

[64] H. Kim, H. Kwon, and K. K. Kim, "Modified cyber kill chain model for multimedia service environments," *Multimedia Tools and Applications*, vol. 78, pp. 3153–3170, 2019.

[65] C. Hankin, P. Malacaria, *et al.*, "Attack dynamics: an automatic attack graph generation framework based on system topology, capec, cwe, and cve databases," *Computers & Security*, vol. 123, pp. 1029–10251, 2022.

[66] F. Mariotti, M. Tavanti, L. Montecchi, and P. Lollini, "Extending a security ontology framework to model capec attack paths and tal adversary profiles," in *2022 18th European Dependable Computing Conference (EDCC)*, pp. 25–32, IEEE, 2022.

[67] A. Mukhopadhyay and S. Jain, "A framework for cyber-risk insurance against ransomware: A mixed-method approach," *International Journal of Information Management*, vol. 74, pp. 1027–1044, 2024.

[68] F. K. Kaiser, L. J. Andris, T. F. Tennig, J. M. Iser, M. Wiens, and F. Schultmann, "Cyber threat intelligence enabled automated attack incident response," in *2022 3rd International Conference on Next Generation Computing Applications (NextComp)*, pp. 1–6, IEEE, 2022.

[69] G. Sakellariou, P. Fouliras, and I. Mavridis, "A methodology for developing & assessing cti quality metrics," *IEEE Access*, 2024.

[70] Y. Li and Q. Liu, "A comprehensive review study of cyber-attacks and cyber security; emerging trends and recent developments," *Energy Reports*, vol. 7, pp. 8176–8186, 2021.

[71] A. Singh and A. Jain, "Study of cyber attacks on cyber-physical system," in *Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT)*, pp. 26–27, 2018.

[72] N. I. Che Mat, N. Jamil, Y. Yusoff, and M. L. Mat Kiah, "A systematic literature review on advanced persistent threat behaviors and its detection strategy," *Journal of Cybersecurity*, vol. 10, no. 1, pp. 23–47, 2024.

[73] F. Li, X. Yan, Y. Xie, Z. Sang, and X. Yuan, "A review of cyber-attack methods in cyber-physical power system," in *2019 IEEE 8th International*

Conference on Advanced Power System Automation and Protection (APAP), pp. 1335–1339, IEEE, 2019.

[74] A. Bendovschi, "Cyber-attacks–trends, patterns and security countermeasures," *Procedia Economics and Finance*, vol. 28, pp. 24–31, 2015.

[75] A. Elitzur, R. Puzis, and P. Zilberman, "Attack hypothesis generation," in *2019 European Intelligence and Security Informatics Conference (EISIC)*, pp. 40–47, IEEE, 2019.

[76] A. J. H. Neto and A. F. P. dos Santos, "Cyber threat hunting through automated hypothesis and multi-criteria decision making," in *2020 IEEE International Conference on Big Data (Big Data)*, pp. 1823–1830, IEEE, 2020.

[77] A. Dimitriadis, N. Ivezic, B. Kulvatunyou, and I. Mavridis, "D4i-digital forensics framework for reviewing and investigating cyber attacks," *Array*, vol. 5, 2020.

[78] F. K. Kaiser, U. Dardik, A. Elitzur, P. Zilberman, N. Daniel, M. Wiens, F. Schultmann, Y. Elovici, and R. Puzis, "Attack hypotheses generation based on threat intelligence knowledge graph," *IEEE Transactions on Dependable and Secure Computing*, 2023.

[79] S. Roy, E. Panaousis, C. Noakes, A. Laszka, S. Panda, and G. Loukas, "Sok: The mitre att&ck framework in research and practice," *arXiv preprint arXiv:2304.07411*, 2023.

[80] A. Georgiadou, S. Mouzakitis, and D. Askounis, "Assessing mitre att&ck risk using a cyber-security culture framework," *Sensors*, vol. 21, no. 9, pp. 3267–2381, 2021.

[81] O. Grigorescu, A. Nica, M. Dascalu, and R. Rughinis, "Cve2att&ck: Bert-based mapping of cves to mitre att&ck techniques," *Algorithms*, vol. 15, no. 9, pp. 314–336, 2022.

[82] P. Rajesh, M. Alam, M. Tahernezhadi, A. Monika, and G. Chanakya, "Analysis of cyber threat detection and emulation using mitre attack framework," in *2022 International Conference on Intelligent Data Science Technologies and Applications (IDSTA)*, pp. 4–12, IEEE, 2022.

[83] V. Mavroeidis, R. Hohimer, T. Casey, and A. Jesang, "Threat actor type inference and characterization within cyber threat intelligence," in *2021 13th International Conference on Cyber Conflict (CyCon)*, pp. 327–352, IEEE, 2021.

[84] H. Al-Mohannadi, I. Awan, and J. Al Hamar, "Analysis of adversary activities using cloud-based web services to enhance cyber threat intelligence," *Service Oriented Computing and Applications*, vol. 14, pp. 175–187, 2020.

[85] H. Almohannadi, I. Awan, J. Al Hamar, A. Cullen, J. P. Disso, and L. Armitage, "Cyber threat intelligence from honeypot data using elasticsearch," in *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, pp. 900–906, IEEE, 2018.

[86] S. Bromander, M. Swimmer, L. P. Muller, A. Jøsang, M. Eian, G. Skjøtskift, and F. Borg, "Investigating sharing of cyber threat intelligence and proposing a new data model for enabling automation in knowledge representation and exchange," *Digital Threats: Research and Practice (DTRAP)*, vol. 3, no. 1, pp. 1–22, 2021.

[87] D. Hermawan, N. G. Novianto, and D. Octavianto, "Development of open source-based threat hunting platform," in *2021 2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, pp. 1–6, IEEE, 2021.

[88] K. Edie, C. Mckee, and A. Duby, "Extending threat playbooks for cyber threat intelligence: A novel approach for apt attribution," in *2023 11th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1–6, IEEE, 2023.

[89] D. Tovarňák, M. Čech, D. Tichỳ, and V. Dohnal, "Observabledb: An inverted index for graph-based traversal of cyber threat intelligence," in *NOMS*

2022-2022 IEEE/IFIP Network Operations and Management Symposium, pp. 1–4, IEEE, 2022.

[90] S. Freeman and M. Bristow, "Consequence is not enough: The role of cyber intelligence in improving cyberattack estimates," *Cyber Security: A Peer-Reviewed Journal*, vol. 7, no. 3, pp. 199–206, 2024.

[91] R. M. Czekster, "Leveraging cyber threat intelligence in smart devices," in *Information Security and Privacy in Smart Devices: Tools, Methods, and Applications*, pp. 71–95, IGI Global, 2023.

[92] J. M. Kizza, "System intrusion detection and prevention," in *Guide to computer network security*, pp. 295–323, Springer, 2024.

[93] A. K. Sangaiah, A. Javadpour, and P. Pinto, "Towards data security assessments using an ids security model for cyber-physical smart cities," *Information Sciences*, vol. 648, pp. 1195–1207, 2023.

[94] O. H. Abdulganiyu, T. Ait Tchakoucht, and Y. K. Saheed, "A systematic literature review for network intrusion detection system (ids)," *International Journal of Information Security*, pp. 1–38, 2023.

[95] R. Chowdhury, S. Sen, A. Goswami, S. Purkait, and B. Saha, "An implementation of bi-phase network intrusion detection system by using real-time traffic analysis," *Expert Systems with Applications*, vol. 224, pp. 1198–1215, 2023.

[96] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, 2019.

[97] L. D. Manocchio, S. Layeghy, W. W. Lo, G. K. Kulatilleke, M. Sarhan, and M. Portmann, "Flowtransformer: A transformer framework for flow-based network intrusion detection systems," *Expert Systems with Applications*, vol. 241, pp. 1225–1254, 2024.

[98] V. Hnamte and J. Hussain, "An extensive survey on intrusion detection systems: Datasets and challenges for modern scenario," in *2021 3rd International Conference on Electrical, Control and Instrumentation Engineering (ICECIE)*, pp. 1–10, IEEE, 2021.

[99] M. F. Umer, M. Sher, and Y. Bi, "Flow-based intrusion detection: Techniques and challenges," *Computers & Security*, vol. 70, pp. 238–254, 2017.

[100] H. Hindy, D. Brosset, E. Bayne, A. Seeam, C. Tachtatzis, R. Atkinson, and X. Bellekens, "A taxonomy and survey of intrusion detection system design techniques, network threats and datasets," 2018.

[101] R. Montasari, F. Carroll, S. Macdonald, H. Jahankhani, A. Hosseinian-Far, and A. Daneshkhah, "Application of artificial intelligence and machine learning in producing actionable cyber threat intelligence," *Digital Forensic Investigation of Internet of Things (IoT) Devices*, pp. 47–64, 2021.

[102] A. K. Balyan, S. Ahuja, U. K. Lilhore, S. K. Sharma, P. Manoharan, A. D. Algarni, H. Elmannai, and K. Raahemifar, "A hybrid intrusion detection model using ega-pso and improved random forest method," *Sensors*, vol. 22, no. 16, pp. 5986–6006, 2022.

[103] J. Asharf, N. Moustafa, H. Khurshid, E. Debie, W. Haider, and A. Wahab, "A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions," *Electronics*, vol. 9, no. 7, pp. 1177–1222, 2020.

[104] S. M. Kasongo and Y. Sun, "A deep learning method with filter based feature engineering for wireless intrusion detection system," *IEEE access*, vol. 7, pp. 38597–38607, 2019.

[105] M. Salem and A.-K. Al-Tamimi, "A novel threat intelligence detection model using neural networks," *IEEE Access*, vol. 10, pp. 131229–131245, 2022.

[106] S. P. RM, P. K. R. Maddikunta, M. Parimala, S. Koppu, T. R. Gadekallu, C. L. Chowdhary, and M. Alazab, "An effective feature engineering for dnn

using hybrid pca-gwo for intrusion detection in iomt architecture," *Computer Communications*, vol. 160, pp. 139–149, 2020.

[107] V. Kumar, D. Sinha, A. K. Das, S. C. Pandey, and R. T. Goswami, "An integrated rule based intrusion detection system: analysis on unsw-nb15 data set and the real time online dataset," *Cluster Computing*, vol. 23, pp. 1397–1418, 2020.

[108] M. A. Alohali, F. N. Al-Wesabi, A. M. Hilal, S. Goel, D. Gupta, and A. Khanna, "Artificial intelligence enabled intrusion detection systems for cognitive cyber-physical systems in industry 4.0 environment," *Cognitive Neurodynamics*, vol. 16, no. 5, pp. 1045–1057, 2022.

[109] M. Guarascio, N. Cassavia, F. S. Pisani, and G. Manco, "Boosting cyber-threat intelligence via collaborative intrusion detection," *Future Generation Computer Systems*, vol. 135, pp. 30–43, 2022.

[110] X. Li, W. Chen, Q. Zhang, and L. Wu, "Building auto-encoder intrusion detection system based on random forest feature selection," *Computers & Security*, vol. 95, pp. 4048–4063, 2020.

[111] M. Asif, S. Abbas, M. Khan, A. Fatima, M. A. Khan, and S.-W. Lee, "Mapreduce based intelligent model for intrusion detection using machine learning technique," *Journal of King Saud University-Computer and Information Sciences*, 2021.

[112] A. Zibak, C. Sauerwein, and A. Simpson, "A success model for cyber threat intelligence management platforms," *Computers & Security*, vol. 111, pp. 167–187, 2021.

[113] M. Sailio, O.-M. Latvala, and A. Szanto, "Cyber threat actors for the factory of the future," *Applied Sciences*, vol. 10, no. 12, pp. 433–458, 2020.

[114] J. A. Guerrero-Saade, "Draw me like one of your french apts—expanding our descriptive palette for cyber threat actors," in *Virus Bulletin Conference, Montreal*, pp. 1–20, 2018.

[115] K. T. Teuwen, "A modular approach to automatic cyber threat attribution using opinion pools," in *2023 IEEE International Conference on Big Data (BigData)*, pp. 3089–3098, IEEE, 2023.

[116] N. Xiao, B. Lang, T. Wang, and Y. Chen, "Apt-mmf: An advanced persistent threat actor attribution method based on multimodal and multilevel feature fusion," *arXiv preprint arXiv:2402.12743*, 2024.

[117] E. Doynikova, E. Novikova, and I. Kotenko, "Attacker behaviour forecasting using methods of intelligent data analysis: A comparative review and prospects," *Information*, vol. 11, no. 3, pp. 168–186, 2020.

[118] F. Skopik and T. Pahi, "Under false flag: using technical artifacts for cyber attack attribution," *Cybersecurity*, vol. 3, pp. 1–20, 2020.

[119] L. Perry, B. Shapira, and R. Puzis, "No-doubt: Attack attribution based on threat intelligence reports," in *2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 80–85, IEEE, 2019.

[120] S. Naveen, R. Puzis, and K. Angappan, "Deep learning for threat actor attribution from threat reports," in *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP)*, pp. 1–6, IEEE, 2020.

[121] U. Noor, Z. Anwar, T. Amjad, and K.-K. R. Choo, "A machine learning-based fintech cyber threat attribution framework using high-level indicators of compromise," *Future Generation Computer Systems*, vol. 96, pp. 227–242, 2019.

[122] V. Legoy, M. Caselli, C. Seifert, and A. Peter, "Automated retrieval of att&ck tactics and techniques for cyber threat reports," *arXiv preprint arXiv:2004.14322*, 2020.

[123] Y. Ghazi, Z. Anwar, R. Mumtaz, S. Saleem, and A. Tahir, "A supervised machine learning based approach for automatically extracting high-level threat

intelligence from unstructured sources," in *2018 International Conference on Frontiers of Information Technology (FIT)*, pp. 129–134, IEEE, 2018.

[124] A. Niakanlahiji, L. Safarnejad, R. Harper, and B.-T. Chu, "Iocminer: Automatic extraction of indicators of compromise from twitter," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 4747–4754, IEEE, 2019.

[125] Z. Iqbal, Z. Anwar, and R. Mumtaz, "Stixgen-a novel framework for automatic generation of structured cyber threat information," in *2018 International Conference on Frontiers of Information Technology (FIT)*, pp. 241–246, IEEE, 2018.

[126] T. Wang and K. P. Chow, "Automatic tagging of cyber threat intelligence unstructured data using semantics extraction," in *2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 197–199, IEEE, 2019.

[127] L. Zongxun, L. Yujun, Z. Haojie, and L. Juan, "Construction of ttps from apt reports using bert," in *2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 260–263, IEEE, 2021.

[128] M. Alkaabi and S. O. Olatunji, "Modeling cyber-attribution using machine learning techniques," in *2020 30th International Conference on Computer Theory and Applications (ICCTA)*, pp. 10–15, IEEE, 2020.

[129] A. Warikoo, "The triangle model for cyber threat attribution," *Journal of Cyber Security Technology*, vol. 5, no. 3-4, pp. 191–208, 2021.

[130] I. Rosenberg, G. Sicard, and E. David, "Deepapt: nation-state apt attribution using end-to-end deep neural networks," in *Artificial Neural Networks and Machine Learning–ICANN 2017: 26th International Conference on Artificial Neural Networks, Alghero, Italy, September 11-14, 2017, Proceedings, Part II 26*, pp. 91–99, Springer, 2017.

[131] H. Jo, Y. Lee, and S. Shin, "Vulcan: Automatic extraction and analysis of cyber threat intelligence from unstructured text," *Computers & Security*, vol. 120, pp. 1027–1040, 2022.

[132] U. Noor, S. Shahid, R. Kanwal, and Z. Rashid, "A machine learning based empirical evaluation of cyber threat actors high level attack patterns over low level attack patterns in attributing attacks," *arXiv preprint arXiv:2307.10252*, 2023.

[133] M. Li, R. Zheng, L. Liu, and P. Yang, "Extraction of threat actions from threat-related articles using multi-label machine learning classification method," in *2019 2nd International Conference on Safety Produce Informatization (IICSPI)*, pp. 428–431, IEEE, 2019.

[134] A. Modi, Z. Sun, A. Panwar, T. Khairnar, Z. Zhao, A. Doupé, G.-J. Ahn, and P. Black, "Towards automated threat intelligence fusion," in *2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC)*, pp. 408–416, IEEE, 2016.

[135] R. R. Ramnani, K. Shivaram, and S. Sengupta, "Semi-automated information extraction from unstructured threat advisories," in *Proceedings of the 10th Innovations in Software Engineering Conference*, pp. 181–187, 2017.

[136] G. Husari, E. Al-Shaer, M. Ahmed, B. Chu, and X. Niu, "Ttpdrill: Automatic and accurate extraction of threat actions from unstructured text of cti sources," in *Proceedings of the 33rd annual computer security applications conference*, pp. 103–115, 2017.

[137] G. Kim, C. Lee, J. Jo, and H. Lim, "Automatic extraction of named entities of cyber threats using a deep bi-lstm-crf network," *International journal of machine learning and cybernetics*, vol. 11, pp. 2341–2355, 2020.

[138] G. Ayoade, S. Chandra, L. Khan, K. Hamlen, and B. Thuraisingham, "Automated threat report classification over multi-source data," in *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pp. 236–245, IEEE, 2018.

[139] K. Kim, J. H. An, and J. Yoo, "A design of il-cytis for automated cyber threat detection," in *2018 International Conference on Information Networking (ICOIN)*, pp. 689–693, IEEE, 2018.

[140] X. Wang, R. Chen, B. Song, J. Yang, Z. Jiang, X. Zhang, X. Li, and S. Ao, "A method for extracting unstructured threat intelligence based on dictionary template and reinforcement learning," in *2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 262–267, IEEE, 2021.

[141] B. I. Kim, N. Kim, S. Lee, H. Cho, and J. Park, "A study on a cyber threat intelligence analysis (cti) platform for the proactive detection of cyber attacks based on automated analysis," in *2018 International Conference on Platform Technology and Service (PlatCon)*, pp. 1–6, IEEE, 2018.

[142] C.-M. Chen, S.-H. Wang, D.-W. Wen, G.-H. Lai, and M.-K. Sun, "Applying convolutional neural network for malware detection," in *2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST)*, pp. 1–5, IEEE, 2019.

[143] Q. Wang, H. Yan, and Z. Han, "Explainable apt attribution for malware using nlp techniques," in *2021 IEEE 21st International Conference on Software Quality, Reliability and Security (QRS)*, pp. 70–80, IEEE, 2021.

[144] Y. Gao, X. Li, H. Peng, B. Fang, and S. Y. Philip, "Hincti: A cyber threat intelligence modeling and identification system based on heterogeneous information network," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 2, pp. 708–722, 2020.

[145] J. Zhao, Q. Yan, J. Li, M. Shao, Z. He, and B. Li, "Timiner: Automatically extracting and analyzing categorized cyber threat intelligence from social data," *Computers & Security*, vol. 95, pp. 1018–1027, 2020.

[146] Z. Zhu and T. Dumitraş, "Featuresmith: Automatically engineering features for malware detection by mining the security literature," in *Proceedings of*

the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 767–778, 2016.

[147] I. Deliu, C. Leichter, and K. Franke, "Extracting cyber threat intelligence from hacker forums: Support vector machines versus convolutional neural networks," in *2017 IEEE International Conference on Big Data (Big Data)*, pp. 3648–3656, IEEE, 2017.

[148] M. Kadoguchi, S. Hayashi, M. Hashimoto, and A. Otsuka, "Exploring the dark web for cyber threat intelligence using machine leaning," in *2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 200–202, IEEE, 2019.

[149] W. Yang and K.-Y. Lam, "Automated cyber threat intelligence reports classification for early warning of cyber attacks in next generation soc," in *Information and Communications Security: 21st International Conference, ICICS 2019, Beijing, China, December 15–17, 2019, Revised Selected Papers 21*, pp. 145–164, Springer, 2020.

[150] M. S. Abu, S. R. Selamat, R. Yusof, and A. Ariffin, "An attribution of cyberattack using association rule mining (arm)," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 2, 2020.

[151] G. Settanni, Y. Shovgenya, F. Skopik, R. Graf, M. Wurzenberger, and R. Fiedler, "Acquiring cyber threat intelligence through security information correlation," in *2017 3rd IEEE International Conference on Cybernetics (CYBCONF)*, pp. 1–7, IEEE, 2017.

[152] F. Avellaneda, E.-H. Alikacem, and F. Jaafar, "Using attack pattern for cyber attack attribution," in *2019 International Conference on Cybersecurity (ICoCSec)*, pp. 1–6, IEEE, 2019.

[153] A. A. Alghamdi and G. Reger, "Pattern extraction for behaviours of multi-stage threats via unsupervised learning," in *2020 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, pp. 1–8, IEEE, 2020.

[154] M. Landauer, F. Skopik, M. Wurzenberger, W. Hotwagner, and A. Rauber, "A framework for cyber threat intelligence extraction from raw log data," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 3200–3209, IEEE, 2019.

[155] L. Qiang, Y. Zeming, L. Baoxu, J. Zhengwei, and Y. Jian, "Framework of cyber attack attribution based on threat intelligence," in *Interoperability, Safety and Security in IoT: Second International Conference, InterIoT 2016 and Third International Conference, SaSeIoT 2016, Paris, France, October 26-27, 2016, Revised Selected Papers 2*, pp. 92–103, Springer, 2017.

[156] F. Jaafar, F. Avellaneda, and E.-H. Alikacem, "Demystifying the cyber attribution: An exploratory study," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing*, pp. 35–40, IEEE, 2020.

[157] P. Kumar, G. P. Gupta, R. Tripathi, S. Garg, and M. M. Hassan, "Dltif: Deep learning-driven cyber threat intelligence modeling and identification framework in iot-enabled maritime transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[158] D. Kim, J. Woo, and H. K. Kim, ""i know what you did before": General framework for correlation analysis of cyber threat incidents," in *MILCOM 2016-2016 IEEE Military Communications Conference*, pp. 782–787, IEEE, 2016.

[159] Y. Kambara, Y. Katayama, T. Oikawa, K. Furukawa, S. Torii, and T. Izu, "Developing the analysis tool of cyber-attacks by using cti and attributes of organization," in *Web, Artificial Intelligence and Network Applications: Proceedings of the Workshops of the 33rd International Conference on Advanced Information Networking and Applications (WAINA-2019) 33*, pp. 673–682, Springer, 2019.

[160] A. Grotto, "Deconstructing cyber attribution: a proposed framework and lexicon," *IEEE Security & Privacy*, vol. 18, no. 1, pp. 12–20, 2019.

[161] Y. Zhou, Y. Tang, M. Yi, C. Xi, and H. Lu, "Cti view: Apt threat intelligence analysis system," *Security and Communication Networks*, vol. 2022, pp. 1–15, 2022.

[162] V. S. C. Putrevu, H. Chunduri, M. A. Putrevu, and S. Shukla, "A framework for advanced persistent threat attribution using zachman ontology," in *Proceedings of the 2023 European Interdisciplinary Cybersecurity Conference*, pp. 34–41, 2023.

[163] L. Qiang, Y. Ze-Ming, L. Bao-Xu, and J. Zheng-Wei, "A reasoning method of cyber-attack attribution based on threat intelligence," *International Journal of Computer and Systems Engineering*, vol. 10, no. 5, pp. 920–924, 2016.

[164] "Threat Group Cards: A Threat Actor Encyclopedia."

[165] T. Casey, "Threat agent library helps identify information security risks," *Intel White Paper*, vol. 2, 2007.

[166] D. D. Protic, "Review of kdd cup '99, nsl-kdd and kyoto 2006+ datasets," *Vojnotehnivki glasnik/Military Technical Courier*, vol. 66, no. 3, pp. 580–596, 2018.

[167] U. S. K. P. M. Thanthrige, J. Samarabandu, and X. Wang, "Machine learning techniques for intrusion detection on public dataset," in *2016 IEEE Canadian conference on electrical and computer engineering (CCECE)*, pp. 1–4, IEEE, 2016.

[168] A. Lavin and S. Ahmad, "Evaluating real-time anomaly detection algorithms–the numenta anomaly benchmark," in *2015 IEEE 14th international conference on machine learning and applications (ICMLA)*, pp. 38–44, IEEE, 2015.

[169] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, and K. Nakao, "Statistical analysis of honeypot data and building of kyoto 2006+ dataset for nids evaluation," in *Proceedings of the first workshop on building analysis datasets and gathering experience returns for security*, pp. 29–36, 2011.

[170] N. Moustafa and J. Slay, "The significant features of the unsw-nb15 and the kdd99 data sets for network intrusion detection systems," in *2015 4th international workshop on building analysis datasets and gathering experience returns for security (BADGERS)*, pp. 25–31, IEEE, 2015.

[171] J. M. Peterson, J. L. Leevy, and T. M. Khoshgoftaar, "A review and analysis of the bot-iot dataset," in *2021 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, pp. 20–27, IEEE, 2021.

[172] M. Ghurab, G. Gaphari, F. Alshami, R. Alshamy, and S. Othman, "A detailed analysis of benchmark datasets for network intrusion detection system," *Asian Journal of Research in Computer Science*, vol. 7, no. 4, pp. 14–33, 2021.

[173] F. P. Shah and V. Patel, "A review on feature selection and feature extraction for text classification," in *2016 international conference on wireless communications, signal processing and networking (WiSPNET)*, pp. 2264–2268, IEEE, 2016.

[174] Y. Shin, K. Kim, J. J. Lee, and K. Lee, "Art: automated reclassification for threat actors based on att&ck matrix similarity," in *2021 world automation congress (WAC)*, pp. 15–20, IEEE, 2021.

[175] S. Goel and B. Nussbaum, "Attribution across cyber attack types: network intrusions and information operations," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1082–1093, 2021.

[176] M. Al-Hawawreh, N. Moustafa, S. Garg, and M. S. Hossain, "Deep learning-enabled threat intelligence scheme in the internet of things networks," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 4, pp. 2968–2981, 2020.

[177] S. Samtani, W. Li, V. Benjamin, and H. Chen, "Informing cyber threat intelligence through dark web situational awareness: The azsecure hacker assets portal," *Digital Threats: Research and Practice (DTRAP)*, vol. 2, no. 4, pp. 1–10, 2021.

[178] K. Li, H. Wen, H. Li, H. Zhu, and L. Sun, "Security osif: Toward automatic discovery and analysis of event based cyber threat intelligence," in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 741–747, IEEE, 2018.

[179] M. R. Rahman, R. Mahdavi-Hezaveh, and L. Williams, "A literature review on mining cyberthreat intelligence from unstructured texts," in *2020 International Conference on Data Mining Workshops (ICDMW)*, pp. 516–525, IEEE, 2020.

[180] K.-P. Saalbach, "Attribution of cyber attacks," *Information Technology for Peace and Security: IT Applications and Infrastructures in Conflicts, Crises, War, and Peace*, pp. 279–303, 2019.

[181] I. Sarhan and M. Spruit, "Open-cykg: An open cyber threat intelligence knowledge graph," *Knowledge-Based Systems*, vol. 233, pp. 1075–1088, 2021.

[182] S. Saeed, S. A. Suayyid, M. S. Al-Ghamdi, H. Al-Muhaisen, and A. M. Almuhaideb, "A systematic literature review on cyber threat intelligence for organizational cybersecurity resilience," *Sensors*, vol. 23, no. 16, pp. 7273–7292, 2023.

[183] N. Sun, M. Ding, J. Jiang, W. Xu, X. Mo, Y. Tai, and J. Zhang, "Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives," *IEEE Communications Surveys & Tutorials*, 2023.

[184] Y. Mei, W. Han, S. Li, X. Wu, K. Lin, and Y. Qi, "A review of attribution technical for apt attacks," in *2022 7th IEEE International Conference on Data Science in Cyberspace (DSC)*, pp. 512–518, IEEE, 2022.

[185] M. Kida and O. Olukoya, "Nation-state threat actor attribution using fuzzy hashing," *IEEE Access*, vol. 11, pp. 1148–1165, 2022.

[186] S. Ejaz, U. Noor, and Z. Rashid, "Visualizing interesting patterns in cyber threat intelligence using machine learning techniques," *Cybernetics and Information Technologies*, vol. 22, no. 2, pp. 96–113, 2022.

[187] M. Ammi, O. Adedugbe, F. M. Alharby, and E. Benkhelifa, "Taxonomical challenges for cyber incident response threat intelligence: a review," *International Journal of Cloud Applications and Computing (IJCAC)*, vol. 12, no. 1, pp. 1–14, 2022.

[188] M. Al-Hawawreh and N. Moustafa, "Explainable deep learning for attack intelligence and combating cyber–physical attacks," *Ad Hoc Networks*, vol. 153, pp. 1033–1043, 2024.

[189] S. Ruohonen, A. Kirichenko, D. Komashinskiy, and M. Pogosova, "Instrumenting opencti with a capability for attack attribution support," *Forensic Sciences*, vol. 4, no. 1, pp. 12–23, 2024.

[190] E. Irshad and A. B. Siddiqui, "Cyber threat attribution using unstructured reports in cyber threat intelligence," *Egyptian Informatics Journal*, vol. 24, no. 1, pp. 43–59, 2023.

[191] J. Kevric, S. Jukic, and A. Subasi, "An effective combining classifier approach using tree algorithms for network intrusion detection," *Neural Computing and Applications*, vol. 28, no. Suppl 1, pp. 1051–1058, 2017.

[192] M. R. Kabir, A. R. Onik, and T. Samad, "A network intrusion detection framework based on bayesian network using wrapper approach," *International Journal of Computer Applications*, vol. 166, no. 4, pp. 13–17, 2017.

[193] D. H. Hagos, A. Yazidi, Ø. Kure, and P. E. Engelstad, "Enhancing security attacks analysis using regularized machine learning techniques," in *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*, pp. 909–918, IEEE, 2017.

[194] M. M. M. Hassan, "Current studies on intrusion detection system, genetic algorithm and fuzzy logic," *arXiv preprint arXiv:1304.3535*, 2013.

[195] S. Duque and M. N. bin Omar, "Using data mining algorithms for developing a model for intrusion detection system (ids)," *Procedia Computer Science*, vol. 61, pp. 46–51, 2015.

[196] B. Agarwal and N. Mittal, "Hybrid approach for detection of anomaly network traffic using data mining techniques," *Procedia Technology*, vol. 6, pp. 996–1003, 2012.

[197] Z. Muda, W. Yassin, M. Sulaiman, and N. Udzir, "K-means clustering and naive bayes classification for intrusion detection," *Journal of IT in Asia*, vol. 4, no. 1, pp. 13–25, 2014.

[198] U. S. Musa, M. Chhabra, A. Ali, and M. Kaur, "Intrusion detection system using machine learning techniques: A review," in *2020 international conference on smart electronics and communication (ICOSEC)*, pp. 149–155, IEEE, 2020.

[199] M. Zaman and C.-H. Lung, "Evaluation of machine learning techniques for network intrusion detection," in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–5, IEEE, 2018.

[200] K. A. Taher, B. M. Y. Jisan, and M. M. Rahman, "Network intrusion detection using supervised machine learning technique with feature selection," in *2019 International conference on robotics, electrical and signal processing techniques (ICREST)*, pp. 643–646, IEEE, 2019.

[201] S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, "A stacking ensemble for network intrusion detection using heterogeneous datasets," *Security and Communication Networks*, vol. 2020, pp. 1–9, 2020.

[202] C. J. Ugochukwu, E. Bennett, and P. Harcourt, *An intrusion detection system using machine learning algorithm*. LAP LAMBERT Academic Publishing, 2019.

[203] H. Alqahtani, I. H. Sarker, A. Kalim, S. M. Minhaz Hossain, S. Ikhlaq, and S. Hossain, "Cyber intrusion detection using machine learning classification

techniques," in *Computing Science, Communication and Security: First International Conference, COMS2 2020, Gujarat, India, March 26–27, 2020, Revised Selected Papers 1*, pp. 121–131, Springer, 2020.

[204] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *Ieee access*, vol. 6, pp. 35365–35381, 2018.

[205] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, pp. 1024–1039, 2020.

[206] M. Azizjon, A. Jumabek, and W. Kim, "1d cnn based network intrusion detection with normalization on imbalanced data," in *2020 international conference on artificial intelligence in information and communication (ICAIIC)*, pp. 218–224, IEEE, 2020.

[207] R. Panigrahi, S. Borah, A. K. Bhoi, M. F. Ijaz, M. Pramanik, Y. Kumar, and R. H. Jhaveri, "A consolidated decision tree-based intrusion detection system for binary and multiclass imbalanced datasets," *Mathematics*, vol. 9, no. 7, pp. 751–786, 2021.

[208] M. Al-Fawa'reh, M. Al-Fayoumi, S. Nashwan, and S. Fraihat, "Cyber threat intelligence using pca-dnn model to detect abnormal network behavior," *Egyptian Informatics Journal*, vol. 23, no. 2, pp. 173–185, 2022.

[209] K.-H. Le, M.-H. Nguyen, T.-D. Tran, and N.-D. Tran, "Imids: An intelligent intrusion detection system against cyber threats in iot," *Electronics*, vol. 11, no. 4, pp. 524–540, 2022.

[210] I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, "Intrudtree: a machine learning based cyber security intrusion detection model," *Symmetry*, vol. 12, no. 5, pp. 754–769, 2020.

[211] "S&P Global Market Intelligence — spglobal.com." https://www.spglobal.com/marketintelligence/en/. [Accessed 19-02-2024].

[212] "List of alternative country names - Wikipedia — en.Wikipedia.org." https://en.Wikipedia.org/wiki/Listofalternativecountrynames. [Accessed 19-02-2024].

[213] "Category:Lists of software - Wikipedia — en.Wikipedia.org." https://en.Wikipedia.org/wiki/Category:Listsofsoftware. [Accessed 19-02-2024].