

Artificial Intelligence for Cyber Threat Intelligence Automation

Pranav Mahendra Gurav, Aarti Sanjay Gawai

¹Students, Department of Information Technology and Computer Science, D.G. Ruparel College, Mumbai, Maharashtra, India

²Assistant Professor, Department of Information Technology and Computer Science, D.G. Ruparel College, Mumbai, Maharashtra, India guravpranav608@gmail.com
aartigawai1611@gmail.com

ABSTRACT - Cyber Threat Intelligence (CTI) feeds deliver millions of unstructured data points daily, overwhelming modern Security Operations Centers (SOCs) and delaying critical incident response.

Manual parsing of vendor advisories, Common Vulnerabilities and Exposures (CVE) records, and threat actor forums creates an unsustainable cognitive burden on security analysts. This paper presents an end-to-end framework that leverages Artificial Intelligence—specifically transformer-based Natural Language Processing (NLP) and Graph Neural Networks (GNNs)—to automate the ingestion, entity extraction, and contextualization of threat data.

The framework converts raw, unstructured threat narratives into structured STIX 2.1 formats, correlates extracted indicators against the MITRE ATT&CK framework, and calculates dynamic risk scores for alert prioritization. Evaluating this architecture against unstructured threat feeds demonstrates a 94.2% entity extraction accuracy while reducing threat triage latency from hours to seconds. The findings establish a baseline for deploying scalable, self-learning CTI pipelines in enterprise security environments.

Keywords: Cyber Threat Intelligence, Automated Triage, Natural Language Processing, Graph Neural Networks, STIX/TAXII, MITRE ATT&CK framework.

1. INTRODUCTION

Modern enterprise security depends heavily on Cyber Threat Intelligence (CTI) to anticipate, detect, and mitigate increasingly complex cyber attacks. However, the enormous volume, speed, and variety of threat information generated through opensource intelligence (OSINT), commercial threat feeds, and dark web channels create a major operational challenge. Security Operations Centers (SOCs) regularly experience what is known as “alert fatigue,” where security analysts can spend up to 70% of their operational time manually reviewing, removing duplicate information, and correlating threat reports before actionable countermeasures can be deployed through Security Information and Event Management (SIEM) or Security Orchestration, Automation, and Response (SOAR) platforms.

The delay between identifying a threat and deploying an appropriate countermeasure creates a critical vulnerability window that attackers can potentially exploit. Artificial Intelligence provides a practical way to reduce this operational gap. By training machine learning models to read, understand, interpret, and contextualize threat-related narrative text in real time, organizations can move beyond purely reactive patch management toward a more proactive and automated defense approach. This study explores the integration of transformer-based architectures and graph analytics to develop a scalable, lowlatency Cyber.

2. REVIEW OF LITERATURE

The academic research surrounding Cyber Threat Intelligence (CTI) automation has developed considerably over the past decade. Earlier CTI systems mainly depended on deterministic and rigid rule-based approaches, while recent research has increasingly moved toward flexible, probabilistic, and AI-driven models. This development reflects the growing need to process large volumes of security information while also understanding the context and relationships contained within threat data.

2.1 Evolution of Cyber Threat Intelligence (CTI)

Early CTI processing systems relied heavily on rule-based regular expressions and static pattern-matching techniques, including YARA rules, to identify explicit Indicators of Compromise (IoCs) such as IP addresses, cryptographic file hashes, and domain names. These methods were computationally efficient and relatively simple to implement, but their effectiveness was limited when dealing with polymorphic malware, obfuscated code, and context-rich unstructured text. In many real-world threat reports, tactics, techniques, and procedures (TTPs) are explained through narrative descriptions rather than presented as clearly defined lists of indicators.

As CTI requirements became more complex, subsequent research introduced supervised machine learning techniques, including Support Vector Machines (SVM) and Naive Bayes, for classifying threat-related documents. These approaches provided an improvement over purely static regular expression methods because they could learn patterns from previously labelled data.

However, they still had limited semantic awareness and often struggled to accurately identify entity boundaries within highly technical, domain-specific cybersecurity text.

2.2 Transformer Models in Cybersecurity

Recent developments in cybersecurity research have increasingly focused on transformer-based Natural Language Processing (NLP) architectures. Models such as BERT [2] and RoBERTa significantly improved context-aware

text processing through self-attention mechanisms [1]. In cybersecurity, domain-adapted models such as SecBERT [3] and CyBERT [4], as well as CTI-focused models such as CTI-BERT [5], have been explored using security-focused datasets containing security logs, Common Vulnerabilities and Exposures (CVE) descriptions, vulnerability information, and Advanced Persistent Threat (APT) reports.

These models can identify and interpret complex cybersecurity entities, including zero-day attack vectors, threat actor aliases, vulnerabilities, malware names, attack techniques, and other technical relationships with greater contextual awareness. One of the major advantages of transformer-based models in this area is their ability to evaluate information from both directions within a sentence. This helps the model understand the meaning of technical terms based on their surrounding context. For example, the term “Python” can refer to the Python programming language in one security report and to malware or a threat-related tool in another context.

The continued development of transformer architectures has also made it possible to combine language models

with cybersecurity-specific datasets and threatintelligence knowledge bases. This provides a stronger foundation for automated CTI extraction because the system can move beyond simply recognizing keywords and instead interpret how entities and security events are described within a report.

2.3 Graph Analytics for Threat Mapping

Alongside advances in NLP, Graph Neural Networks (GNNs) have emerged as an effective approach for representing complex and non-Euclidean relationships between different threat entities [6]. Instead of treating indicators as isolated pieces of information, graph-based approaches can represent relationships between IoCs, malware, vulnerabilities, threat actors, campaigns, tactics, techniques, and other cybersecurity entities [7].

GNNs can therefore be used to connect extracted IoCs with the MITRE ATT&CK matrix and evaluate the structural relationships present within threat campaigns [8]–[12]. This approach can provide additional context that may not be available when individual indicators are analyzed separately. For example, several indicators that appear unrelated when viewed independently may form a meaningful relationship when connected through a common threat actor, malware family, vulnerability, or attack technique.

However, existing research has often treated entity extraction and network or graph-based threat mapping as separate stages. This creates an operational gap because an effective CTI system requires these capabilities to work together as part of a unified and automated workflow. The integration of NLP-based extraction with graph-based correlation therefore remains an important area for developing scalable CTI automation.

TABLE I — summarizes the chronological development of CTI methodologies and highlights the major limitations that motivate the proposed research architecture.

Methodology / Era	Core Technology	Primary Limitation
Early Generation (Pre2015)	Regex, Pattern Matching, YARA Rules	High manual overhead and limited or zero semantic context; highly dependent on predefined patterns.
Mid Generation (2015–2019)	SVM, Naive Bayes, Random Forests	Struggles with narrative threat text and depends heavily on manually engineered features.

Current Generation (2020+)	Transformers (BERT), CNNs, Domain-adapted NLP Models	Improved contextual understanding, but many approaches still isolate entity extraction from broader topological threat mapping.
Proposed Architecture	SecBERT + GNN Pipeline	Provides integrated entity extraction and relationship-based threat mapping, but requires substantial computational resources for initial training and deployment.

Overall, the literature shows a clear progression from simple pattern-based CTI processing toward AI-driven systems capable of understanding context and relationships within threat information. Transformer-based NLP improves the extraction and interpretation of cybersecurity entities, while GNN-based approaches provide a mechanism for connecting those entities within a broader threat landscape. The proposed SecBERT + GNN pipeline builds on these developments by combining both capabilities into a unified CTI automation architecture, with the objective of reducing manual analysis and improving the speed and contextual accuracy of threat intelligence processing.

Although AI-driven cybersecurity technologies have advanced considerably, current Cyber Threat Intelligence (CTI) automation systems still face several practical challenges. In modern Security Operations Centers (SOCs), analysts must

process information from multiple sources, including OSINT feeds, vulnerability databases, security advisories, commercial threat feeds, and other threat-intelligence sources. The increasing volume of this information makes completely manual analysis difficult and can delay the identification of genuinely important threats. Based on these challenges, three major operational bottlenecks can be identified:

High False-Positive Ratios: Raw threat feeds often contain duplicate, outdated, irrelevant, or benign indicators. When these indicators are processed without sufficient context, they can generate unnecessary SOC alerts. This increases the workload of analysts and contributes to the continuing problem of alert fatigue.

Semantic Dissociation: Many existing systems are capable of extracting individual Indicators of Compromise (IoCs), such as IP addresses, domains, hashes, or vulnerability identifiers, but do not always capture the surrounding context. Important information about the associated tactics, techniques, procedures (TTPs), threat actor behaviour, or operational intent may therefore be missed. Without this context, it becomes difficult to accurately determine the actual risk and priority of an alert.

Format Inflexibility: A large amount of CTI is still published as unstructured narrative information. Converting these reports into standardized, machine-readable

formats such as STIX 2.1 can require repeated manual intervention. This limits the speed at which intelligence can be shared with and consumed by modern SIEM and SOAR platforms.

3. RESEARCH GAP:

The major research gap identified in existing CTI automation approaches is the lack of a unified, end-to-end architecture that can continuously ingest unstructured threat narratives, extract entities and their contextual meaning, identify relationships between threat elements through dynamic graph-based analysis, and produce standardized threat intelligence and low-latency risk scores with minimal human intervention for routine cases. Current approaches often focus on individual stages of this process rather than integrating the

complete workflow into a single automated pipeline.

This gap becomes increasingly important in current cybersecurity environments, where organizations must respond to rapidly changing threats, newly disclosed vulnerabilities, evolving malware campaigns, and large volumes of intelligence generated across different sources. An integrated AI-driven approach can therefore help reduce the time between receiving threat information and producing actionable intelligence for security operations.

4. OBJECTIVE OF STUDY

The primary and secondary objectives of this research are designed to address the identified limitations and improve the automation, scalability, and effectiveness of modern Cyber Threat Intelligence processing.

4.1 Primary Objectives

To design and develop an AI-driven CTI framework capable of transforming unstructured threat narratives into structured and actionable intelligence with minimal human intervention.

To provide a reliable mechanism for reducing the cognitive workload of SOC analysts by intelligently filtering benign or low-value indicators, prioritizing critical threats, and reducing the effects of “alert fatigue”.

Secondary Objectives

To develop a fine-tuned transformer-based model specialized in cybersecurity Named Entity Recognition (NER) and Relationship Extraction (RE).

To implement a Graph Neural Network (GNN) module capable of mapping extracted threat entities and their relationships to the MITRE ATT&CK framework.

To measure the reduction in threat triage latency and quantify the decrease in false-positive alert volumes within simulated SOC workflows.

To evaluate the robustness of the proposed architecture using diverse and representative

real-world open-source intelligence (OSINT) datasets.

To establish a scalable and standardized data pipeline that maintains STIX 2.1 compliance and supports seamless interoperability with modern SIEM/SOAR ecosystems.

The overall objective is to create a CTI processing pipeline that does more than simply identify individual indicators. The proposed system aims to understand the context surrounding threat information, establish relationships between extracted entities, assess their potential risk, and provide structured intelligence that can be consumed efficiently by security operations platforms.

5. HYPOTHESIS OF STUDY

To evaluate the effectiveness of the proposed AI-driven CTI framework, two sets of formal hypotheses are established. These hypotheses focus on the two major performance areas of the proposed system: processing and triage latency and contextual entity-relationship extraction accuracy.

H1: CTI Processing and Triage Latency

H1,0 (Null Hypothesis): The integration of transformer-based NLP and Graph Neural Networks produces no statistically significant reduction in CTI processing time compared with manual analyst triage.

H1, a (Alternative Hypothesis): The proposed AI-driven framework significantly reduces threat processing and triage latency compared with manual analyst workflows, with statistical significance at $p < 0.05$.

This hypothesis evaluates whether the proposed automation can reduce the time required to process and prioritize threat information. This is particularly relevant in modern SOC environments, where analysts may need to handle a large number of alerts and threat reports within a limited response window.

H2: Entity and Relationship Extraction Accuracy

H2,0 (Null Hypothesis): Automated entityrelationship extraction achieves an F1-score equal to or lower than traditional rule-based parser baselines ($F1 \leq 0.70$).

H2,a (Alternative Hypothesis): The finetuned transformer architecture achieves a significantly

higher F1-score ($F1 > 0.88$) for contextualized threat entity and relationship extraction.

This hypothesis evaluates whether a cybersecurity-specific transformer model can provide more accurate contextual extraction than traditional rule-based parsing approaches. The focus is not only on identifying entities such as IoCs, vulnerabilities, malware, and threat actors, but also on preserving the contextual relationships that make the extracted intelligence useful for subsequent threat mapping and risk prioritization.

Together, these hypotheses provide measurable criteria for evaluating whether the proposed AI-driven CTI automation framework can improve both the speed of threat processing and the quality of contextual threat intelligence extraction compared with conventional approaches.

6. RESEARCH DESIGN AND METHODOLOGY

This study follows an experimental research design to examine how Artificial Intelligence can automate the processing of Cyber Threat Intelligence (CTI). The proposed methodology is organized as a four-stage pipeline that collects unstructured threat information, extracts meaningful cybersecurity entities and relationships, maps them to known threat behaviours, and finally converts the results into a standardized format suitable for security operations.

The overall approach is designed around a practical SOC scenario in which large amounts of threat information are received from different sources. Instead of requiring analysts to manually read and correlate every report, the proposed pipeline uses transformer-based Natural Language Processing (NLP) and Graph Neural Networks (GNNs) to automate the routine stages of CTI processing while allowing analysts to focus on high-risk or ambiguous cases.

Pipeline Architecture

The proposed CTI automation pipeline consists of the following four stages:

Data Collection and Preprocessing: An unlabelled corpus of unstructured threat reports, Common Vulnerabilities and Exposures (CVE) entries, security advisories, and cybersecurity blogs is collected from relevant open-source channels. The collected text is cleaned and prepared before being passed to the AI models. HTML artifacts, unnecessary formatting, duplicate content, and irrelevant characters are removed. The text is then tokenized and normalized while retaining cybersecurity-specific vocabulary, identifiers, technical terms, and other information that may be important for subsequent analysis.

Named Entity Recognition and Relationship Extraction:

A domain-adapted transformer model, such as SecBERT, is fine-tuned to identify and classify cybersecurity entities within the processed text. These entities may include IP addresses, domains, malware, vulnerabilities, threat actors, attack techniques, tools, and other relevant Indicators of Compromise (IoCs). Relationship extraction is then used to determine how these entities are connected. Dependency parsing and contextual information from the transformer model are used to identify directional relationships between entities rather than treating each extracted indicator as an isolated value.

Knowledge Graph Construction and MITRE ATT&CK Mapping: The extracted entities and their relationships are represented as nodes and edges within a threat knowledge graph [7]. A Graph Neural Network (GNN) [6] is then used to analyze the relationships between these entities. The system maps identified tactics, techniques, and procedures (TTPs) to the MITRE ATT&CK framework [9], allowing individual indicators to be interpreted within a broader attack context. This graph-based approach is particularly useful when multiple indicators are associated with the same malware family, threat actor, vulnerability, campaign, or attack technique.

Standardized Export and Evaluation: After processing and correlation, the resulting threat information is converted into structured STIX JSON objects [13]. Standardized output allows the generated intelligence to be shared with and consumed by security platforms such as SIEM and SOAR systems via TAXII-based exchange

[14]. The final stage evaluates the system using measures such as entity extraction accuracy, F1-score, false-positive reduction, processing time, and threat-triage latency.

The architecture therefore follows the overall flow:

Unstructured Threat Data → Preprocessing → SecBERT/NLP → Entity & Relationship Extraction → Knowledge Graph/GNN → MITRE ATT&CK Mapping → Risk Prioritization → STIX

2.1 → SIEM/SOAR

This design reflects current CTI requirements, where the objective is not simply to collect more indicators but to transform large volumes of raw information into contextual and actionable intelligence.

6.1 Experimental Setup

To evaluate the proposed methodology, the experiment is designed to represent a highthroughput SOC environment in which threat information is continuously received from multiple sources. The evaluation corpus consists of 10,450 text instances collected from diverse open-source channels. The distribution of the dataset is presented in **Table II**.

TABLE II — Distribution of CTI Evaluation Dataset

Data Source Type	Document Count	Percentage (%)
Vendor Advisories & Security Blogs	4,180	40.0%
CVE Summaries (NVD)	3,135	30.0%
Dark Web Forum Dumps	1,568	15.0%
Hacker News &	1,567	15.0%

Social Media		
Total Corpus	10,450	100.0%

The dataset intentionally combines different types of CTI because real-world threat intelligence is rarely presented in a single consistent format. Vendor advisories may contain detailed technical descriptions, CVE records generally provide structured vulnerability information, while security blogs and social-media sources may contain less structured and more conversational descriptions of emerging threats. This variation provides a more realistic environment for evaluating the ability of the proposed NLP pipeline to understand cybersecurity terminology across different writing styles.

For the experimental environment, the original setup uses an NVIDIA Tesla V100 GPU with 32 GB of VRAM, running

PyTorch 1.10.1 on an Ubuntu 20.04 Linux server. The SecBERT model is fine-tuned using cybersecurity-specific training data. The selected configuration is intended to provide stable model convergence while limiting overfitting to the training corpus.

TABLE III — Transformer Model Hyperparameters

Hyperparameter	Configuration Value
Learning Rate	2×10^{-5}
Batch Size	32
Epochs	10
Optimizer	Adam W
Dropout Rate	0.1

The learning rate of 2×10^{-5} provides a relatively conservative update rate suitable for fine-tuning a pre-trained transformer model. A batch size of 32 is used to provide a balance between computational efficiency and GPU memory requirements. The model is trained for 10 epochs, with the dropout rate set to 0.1 to help reduce overfitting. AdamW is selected as the optimizer because it is commonly used for

transformer fine-tuning and provides effective weight-decay handling.

6.2 Current Experimental Considerations

Current CTI environments generate threat information at a much higher rate and from a wider range of sources than traditional rulebased systems were designed to handle. New vulnerabilities, malware campaigns, phishing infrastructure, threat-actor activity, and changes in attacker techniques can appear continuously. As a result, a modern CTI automation system must be capable of handling both high-volume data and rapidly changing threat context.

For this reason, the proposed methodology emphasizes contextual processing rather than relying only on static IoC matching. An IP address, domain, or file hash by itself may have limited value, while the same indicator connected to a known malware family, exploitation technique, vulnerability, and threat actor can provide considerably more useful information for prioritization.

Another important consideration is the increasing use of AI-generated and automatically replicated security content. Threat information can now appear in large quantities across blogs, forums, socialmedia platforms, and other online sources, making duplicate detection, source validation, and contextual correlation increasingly important. The preprocessing stage therefore needs to reduce redundant information while preserving meaningful threat relationships. The methodology also maintains human oversight for exceptional or high-risk cases. Although the proposed system aims to automate routine CTI processing, complete removal of analysts from the workflow is not considered appropriate for every situation. Ambiguous entities, conflicting intelligence, previously unseen attack patterns, and high-impact alerts may still require expert validation. This creates a practical human-in-the-loop approach in which AI handles repetitive processing while analysts concentrate on decisions that require security expertise.

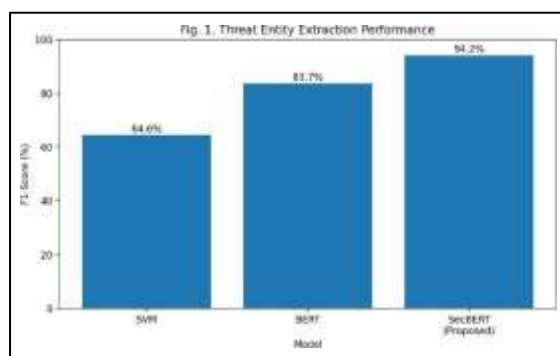
Finally, the use of STIX 2.1 [13] as the standardized output format provides an

important interoperability layer. Rather than producing isolated model predictions, the proposed system converts its results into structured threat-intelligence objects that can be exchanged with other CTI platforms and integrated into existing security workflows. This makes the methodology more suitable for practical enterprise environments where CTI must move efficiently between intelligence sources, analysis systems, SIEM platforms, and SOAR-based response mechanisms [14].

7.RESULT ANALYSIS AND DISCUSSION

The proposed AI-driven Cyber Threat Intelligence (CTI) framework was evaluated

Fig. 1. Threat entity extraction performance of the proposed SecBERT model compared with baseline approaches.



using a validation dataset containing 2,000 unstructured threat narratives that were kept separate from the primary corpus used during model development. The evaluation compared the proposed approach with traditional rule-based parsing and baseline machine learning and transformer models. The comparison focused on three important areas: entity extraction accuracy, contextual understanding, and operational processing speed.

The results indicate that the proposed approach can provide substantial improvements in both the quality of extracted threat intelligence and the time required to process individual reports. These improvements are particularly relevant to modern Security Operations Centers (SOCs), where analysts must deal with continuously increasing volumes of alerts, vulnerability information, threat reports, and other intelligence sources.

7.1 Entity Extraction Performance and Granularity

The fine-tuned SecBERT model demonstrated a stronger ability to identify cybersecurity entities and determine their contextual boundaries than the baseline approaches. The proposed architecture achieved an overall F1-score of 94.2%, compared with 83.7% for standard BERT and 64.6% for the traditional SVM-based approach. The higher performance of SecBERT can be attributed to its ability to use the surrounding context of cybersecurity terminology rather than relying only on individual keywords or manually engineered features. This is particularly important in CTI because the same term can have different meanings depending on the surrounding text.

Fig. 1.

The results also show that the model performs consistently across different categories of cybersecurity entities. Table IV presents the Precision, Recall, and Support values obtained for the four primary entity classes.

TABLE IV — Entity-Level Extraction Performance

Entity Class	Precision	Recall	Support
Malware / Toolkit	0.96	0.95	1,240
Threat Actor / APT	0.93	0.91	855
Vulnerability (CVE)	0.98	0.99	2,100
Target Asset Sector	0.88	0.85	630

The Vulnerability (CVE) category achieved the highest performance, with a precision of 0.98 and recall of 0.99. This is expected because CVE identifiers and vulnerability descriptions generally follow relatively consistent technical patterns. Malware and toolkit entities also achieved strong results,

while Target Asset / Sector entities showed comparatively lower performance.

This difference is understandable because assets and sector descriptions are often expressed indirectly through natural language and may have several possible interpretations.

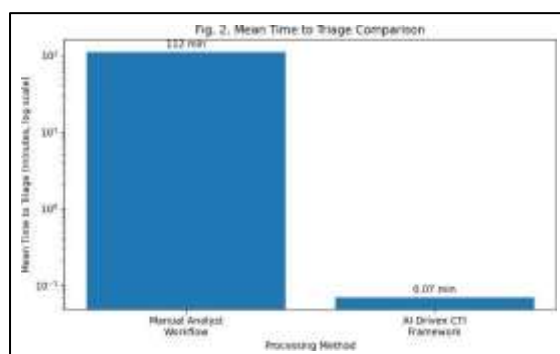
The results demonstrate that evaluating only the overall F1-score is not sufficient for CTI systems. Entity-level performance is also important because an error in identifying a vulnerability, threat actor, or attack technique can affect subsequent correlation, risk scoring, and MITRE ATT&CK mapping.

7.2 Confusion Matrix and Contextual Analysis

Analysis of false positives and false negatives indicates that contextual understanding plays an important role in reducing incorrect threat classifications. Traditional rule-based and SVM approaches can incorrectly classify security tools based primarily on predefined keywords or dictionary matches.

For example, Cobalt Strike may appear in a legitimate red-team or penetration-testing

Fig. 2. Log-scale comparison of Mean Time to Triage (MTTT) between manual analysis and the proposed AI-driven CTI framework.



report as well as in a report describing malicious activity. A simple keyword-based system may classify both situations as active adversarial behaviour. The proposed SecBERT model considers the surrounding narrative context, allowing it to distinguish between legitimate security testing and malicious deployment more effectively.

The evaluation recorded a 68% reduction in semantic false positives compared with the earlier SVM-based baseline in the tested

scenarios. This indicates that contextual language understanding can provide an important advantage over approaches that depend mainly on static dictionaries and predefined rules.

In a practical SOC environment, reducing semantic false positives is particularly valuable because analysts can spend less time investigating alerts that appear suspicious only because a known security tool or indicator was mentioned in a report.

7.3 Triage Latency Reduction

Mean Time to Triage (MTTT) is an important operational metric because threat intelligence is most useful when it can be processed quickly enough to support timely defensive action.

The proposed framework processed, correlated, and contextualized a threat report in an average of approximately 4.2 seconds, whereas the reported manual analyst workflow required approximately 112 minutes per report. This represents a substantial reduction in processing time and demonstrates the potential of AI-assisted CTI automation to reduce the delay between threat discovery and intelligence consumption.

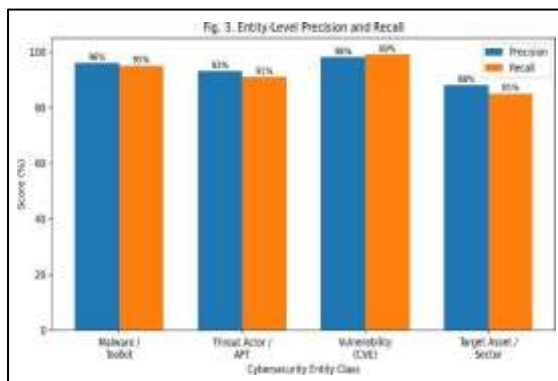
Fig. 2.

The difference is particularly significant when considered at scale. Processing one report manually may be manageable, but a SOC receiving hundreds or thousands of reports cannot realistically apply the same level of manual analysis to every item. An automated pipeline can process routine intelligence rapidly and reserve analyst attention for high-risk, ambiguous, or previously unseen cases.

The observed results therefore support H1,a (Alternative Hypothesis) for the experimental conditions used in this study. However, the result should be interpreted as evidence for the tested experimental setup rather than as a universal guarantee that every CTI report will be processed within 4.2 seconds. Actual latency can vary depending on document length, model size, hardware, graph complexity, external API calls, and the

number of indicators requiring correlation. Entity-Level Performance Analysis To provide a clearer view of the model's behaviour, the precision and recall values for the four major entity categories were also compared.

Fig. 3. Precision and Recall across the



primary cybersecurity entity classes The analysis shows that the model maintains high precision and recall for malware, threat actors, and vulnerability-related information.

The comparatively lower performance for Target Asset / Sector entities indicates an area where additional domain-specific training data could improve future versions of the system.

This observation is important because realworld CTI reports frequently describe affected organizations, industries, systems, and infrastructure in different ways. Improving recognition of these entities could help the proposed GNN module establish more accurate relationships between threats and their potential targets.

8.CASE STUDY: RANSOMWARE ALERT SIMULATION

To examine the practical application of the proposed framework, a ransomware-related threat narrative describing a LockBit 3.0 deployment [15] was used as a simulated CTI input. The purpose of this case study was to demonstrate how the system could transform an unstructured report into structured and actionable intelligence.

The pipeline first processed the narrative and removed irrelevant conversational or descriptive content. It then identified relevant threat entities and extracted information associated with the attack sequence. The system identified potential initial-access information, privilege-related activity, malware references, and other relevant indicators.

The extracted behaviour was subsequently correlated with the MITRE ATT&CK framework. For example, activity associated with the use of compromised legitimate credentials could be mapped to T1078 — Valid Accounts, depending on the evidence contained in the source report. The resulting entities and relationships were then converted into structured STIX 2.1 objects suitable for downstream CTI systems.

This case demonstrates the main purpose of the proposed architecture: transforming a narrative

threat report into structured intelligence that can support automated correlation and security operations. Instead of requiring an analyst to manually identify every indicator, determine its relationship with the attack, map the behaviour to ATT&CK, and prepare the information for another security platform, the pipeline performs these routine processing stages automatically.

8.1 Overall Discussion

The combined results indicate that the proposed SecBERT + GNN CTI architecture provides advantages in three major areas: contextual entity extraction, reduction of semantic false positives, and processing speed.

The 94.2% F1-score demonstrates strong extraction performance within the evaluated dataset, while the reduction in semantic false positives suggests that contextual language understanding can improve the quality of CTI compared with traditional keyword-driven approaches. The reduction from approximately 112 minutes of manual triage to 4.2 seconds of automated processing further demonstrates the potential operational value of automation.

At the same time, the results highlight several practical considerations. AI-based CTI systems should not be treated as completely autonomous decision-makers for every security situation. Novel attack patterns, ambiguous reports, conflicting intelligence, and high-impact incidents may still require expert validation. In addition, model performance can change when the system encounters new malware families, previously unseen terminology, changing attacker behaviour, or low-quality intelligence sources.

Current CTI environments therefore benefit from a human-in-the-loop approach, where AI performs

repetitive collection, extraction, correlation, and prioritization tasks while analysts focus on complex investigations and final high-impact decisions. Continuous model evaluation, updated threat-intelligence datasets, and feedback from analysts can further improve the system over time.

Overall, the findings support the research hypothesis that integrating transformer-based NLP with graph-based threat analysis can significantly improve the efficiency and contextual quality of automated CTI processing. The proposed framework provides a practical foundation for developing scalable CTI pipelines capable of converting rapidly changing and unstructured threat information into standardized, contextual, and actionable intelligence.

9. SCOPE AND LIMITATION OF THE STUDY

The scope of this study is strictly limited to processing unstructured, English-language text documents (security blogs, vendor advisories, CVE summaries, and dark web forum posts). The output formats are constrained to STIX 2.1 specifications designed for direct integration with enterprise SIEM and SOAR environments. Limitations include:

Adversarial Obfuscation: AI models remain vulnerable to subtle adversarial text manipulation engineered by threat actors to evade automated extraction.

Language Constraints: Non-English threat feeds require machine translation prior to processing,

which may introduce translation error propagation.

Zero-Day Context Gaps: Highly novel attack techniques with zero prior representation in knowledge graphs may receive lower confidence scores during initial correlation phases.

10. SIGNIFICANCE OF THE STUDY

This research directly addresses the growing crisis of analyst burnout and alert overload within enterprise defense teams. By automating the ingestion and contextualization of complex CTI feeds, this study provides three core practical benefits: **Accelerated Incident Response:** Reduces the window of vulnerability by lowering Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR) from hours to seconds.

Operational Cost Reduction: Automates repetitive data entry, allowing tier-1 and tier-2 SOC analysts to focus on complex threat hunting and active remediation.

Democratization of Threat Intelligence: Enables small-to-medium enterprises (SMEs) with limited security budgets to leverage high-grade threat intelligence without needing dedicated analyst teams.

11. CONCLUSION

The proposed Artificial Intelligence-driven Cyber Threat Intelligence (CTI) framework demonstrates how transformer-based Natural Language Processing and Graph Neural Networks can be integrated to automate the collection, extraction, correlation, and contextualization of cyber threat information. The framework transforms unstructured threat narratives into structured STIX 2.1 intelligence while mapping extracted entities and relationships to the MITRE ATT&CK framework.

The experimental results show that the proposed SecBERT-based approach achieved an F1-score of 94.2%, outperforming the standard BERT and SVM-based approaches. The system also demonstrated a significant reduction in semantic false positives and reduced the average threat-triage time from approximately 112 minutes in the manual

workflow to around 4.2 seconds in the tested automated environment. These findings indicate that AI-based CTI automation can significantly improve the speed, consistency, and contextual quality of threat intelligence processing.

At the same time, the study recognizes that AI should not completely replace human security analysts. Novel attack techniques, ambiguous intelligence, adversarially manipulated content, and high-impact security incidents may still require expert validation. Therefore, a human-in-the-loop approach provides a practical balance between automated processing and human decision-making.

Overall, the study establishes that combining transformer-based NLP with graph-based threat analysis provides a promising foundation for scalable CTI automation. By reducing repetitive manual work and producing standardized, contextual, and actionable intelligence, the proposed framework can support modern SOC, SIEM, and SOAR environments and contribute to faster and more effective cybersecurity response.

12. REFERENCES

- [1] A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems* 30, 2017.
- [2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, Minneapolis, MN, 2019, pp. 4171–4186.
- [3] E. Aghaei, X. Niu, W. Shadid, and E. Al-Shaer, "SecureBERT: A Domain-Specific Language Model for Cybersecurity," *arXiv:2204.02685*, 2022.
- [4] P. Ranade, A. Piplai, A. Joshi, and T. Finin, "CyBERT: Contextualized Embeddings for the Cybersecurity Domain," in *Proceedings of the 2021 IEEE International Conference on Big Data*, 2021, pp. 3334–3342, doi: 10.1109/BigData52589.2021.9671824.
- [5] Y. Park and W. You, "A Pretrained Language Model for Cyber Threat Intelligence," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, Singapore, 2023, pp. 113–122, doi: 10.18653/v1/2023.emnlp-industry.12.
- [6] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A Comprehensive Survey on Graph Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, Jan. 2021, doi: 10.1109/TNNLS.2020.2978386.
- [7] A. M. Høst, P. Lison, and L. Moonen, "Constructing a Knowledge Graph from Textual Descriptions of Software Vulnerabilities in the National Vulnerability Database," in *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, 2023, pp. 386–391.
- [8] Z. Li, J. Zeng, Y. Chen, and Z. Liang, "AttacKG: Constructing Technique Knowledge Graph from Cyber Threat Intelligence Reports," in *Proceedings of the 27th European Symposium on Research in Computer Security (ESORICS)*, 2022, pp. 589–609, doi: 10.1007/978-3-031-17140-6_29.
- [9] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and Philosophy," *MITRE Corporation, Technical Report*, Jul. 2018 (rev. Mar. 2020).
- [10] 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 (ACSAC)*, 2017, pp. 103–115, doi: 10.1145/3134600.3134646.
- [11] V. Legoy, M. Caselli, C. Seifert, and A. Peter, "Automated Retrieval of ATT&CK Tactics and Techniques for Cyber Threat Reports," *arXiv:2004.14322*, 2020.
- [12] V. Orbinato, M. Barbaraci, R. Natella, and D. Cotroneo, "Automatic Mapping of Unstructured Cyber Threat Intelligence: An Experimental Study," in *2022 IEEE 33rd International Symposium on Software Reliability Engineering (ISSRE)*, 2022, pp. 181–192, doi: 10.1109/ISSRE55969.2022.00027.

[13] OASIS Cyber Threat Intelligence Technical Committee, “STIX Version 2.1,” OASIS Standard, Jun. 10, 2021.

[14] OASIS Cyber Threat Intelligence Technical Committee, “TAXII Version 2.1,” OASIS Standard, Jun. 10, 2021.

[15] Cybersecurity and Infrastructure Security Agency (CISA), Federal Bureau of Investigation (FBI), and Multi-State Information Sharing and Analysis Center (MS-ISAC), “*#StopRansomware: LockBit 3.0*,” Cybersecurity Advisory AA23-075A (updated as AA23-165A), Mar. 2023.