HINTELL: An Intelligent Heterogeneous Graph and LLM-Driven Framework for Automated Cyber Threat Intelligence Analysis

Report submitted to

Indian Institute of Technology, Kharagpur

for the award of the degree

B.Tech. (Hons) in Mechanical Engineering

&

M.Tech. in Manufacturing Science and Engineering

by

Sambit Kumar Sahoo
Roll No: 22ME31052

Under the supervision of

Prof. Akhilesh Kumar

DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR
November 2025

DECLARATION

I certify that

1. the work contained in this report is original and has been done by me under the guidance of my supervisor(s);
2. the work has not been submitted to any other Institute for any degree or diploma;
3. I have followed the guidelines provided by the Institute in preparing the report;
4. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute;
5. whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the report and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

_____

Signature of the Student
Sambit Kumar Sahoo

CERTIFICATE

This is to certify that the B.Tech Project entitled **"HINTELL: An Intelligent Heterogeneous Graph and LLM-Driven Framework for Automated Cyber Threat Intelligence Analysis"** submitted by **Mr. Sambit Kumar Sahoo** to the **Indian Institute of Technology, Kharagpur, India**, is a record of bona fide project work carried out by him under my supervision and guidance, and is worthy of consideration for the award of the degree of **Bachelor of Technology in Mechanical Engineering** of the Institute.

Supervisor	**Co-Supervisor (if any)**

Date:

Acknowledgement

I express my deepest sense of gratitude to **Prof. Akhilesh Kumar**, Industrial and Systems Engineering Department, Indian Institute of Technology Kharagpur, for his invaluable guidance, encouragement, and continuous support throughout the course of this project. His insightful suggestions, patient supervision, and constant motivation have been instrumental in shaping the direction and successful completion of this work.

I am especially thankful for his time, constructive feedback, and deep technical discussions that helped me refine my understanding of both artificial intelligence methodologies and their practical application in cyber threat intelligence. Working under his mentorship has been a truly enriching learning experience.

I would also like to extend my appreciation to all faculty members and peers who provided helpful discussions and a collaborative environment during the project. Their valuable inputs and feedback have contributed significantly to improving the overall quality of this work.

Finally, I am grateful to the Indian Institute of Technology Kharagpur for providing the necessary infrastructure, research environment, and academic resources that made this work possible.

Sambit Kumar Sahoo
Roll No.: 22ME31052
Department of Mechanical Engineering
Indian Institute of Technology Kharagpur

Contents

ABSTRACT

The increasing sophistication of cyber threats necessitates the development of intelligent, automated systems capable of analyzing large volumes of heterogeneous threat data. This project presents **HINTELL**, an integrated framework that leverages heterogeneous graph neural networks (HGNNs) and large language models (LLMs) to enhance cyber threat intelligence (CTI) generation, interpretation, and prediction.

The proposed system begins with a fine-tuned entity extraction model that identifies Indicators of Compromise (IOCs) such as IPs, domains, malware, vulnerabilities, and threat actors from unstructured threat reports. Extracted entities are normalized and organized into a **Heterogeneous Information Network (HIN)**, where diverse node and relation types capture the multi-faceted nature of cyber threat ecosystems. Meta-path based similarity computation and PathSim analysis are then employed to uncover latent semantic patterns and behavioral correlations across entities. These enriched relations are utilized to train a graph neural network that learns robust node embeddings for downstream tasks such as link prediction and node classification.

Finally, an **LLM driven interpretation layer** contextualizes the GNN predictions, providing human-readable explanations, behavioral clustering insights, and actionable threat intelligence recommendations. The framework demonstrates a scalable and explainable approach for integrating symbolic reasoning (via meta-paths) with neural representations (via GNNs) and language understanding (via LLMs).

This work contributes a holistic AI-powered CTI modeling pipeline that enhances situational awareness, supports early threat detection, and facilitates proactive cyber defense.

Chapter 1 Introduction

In recent years, the volume and sophistication of cyberattacks have increased at an unprecedented rate. Modern threat actors employ diverse tactics and malware infrastructures that continuously evolve to evade conventional defense mechanisms. Consequently, understanding and predicting these threats require intelligent systems capable of extracting, linking, and reasoning over large volumes of unstructured cyber threat data.

Cyber Threat Intelligence (CTI) has emerged as a strategic approach for organizations to gain insights into adversaries' behaviors, tools, and infrastructures. CTI involves the collection, analysis, and dissemination of Indicators of Compromise (IOCs), such as IP addresses, domains, malware names, file hashes, and vulnerabilities that reveal the footprints of cyber incidents. However, manually analyzing and correlating these indicators from numerous threat reports, advisories, and databases is both time-consuming and error-prone. This creates the need for an automated, intelligent framework that can extract structured knowledge and derive meaningful relationships among threat entities.

1.1 Cyber Threat Intelligence (CTI): Background and Significance

Cyber Threat Intelligence involves gathering and analyzing **Indicators of Compromise (IOCs)** such as malicious IP addresses, domains, malware, vulnerabilities, and threat actors. When linked meaningfully, these entities provide insight into the intent, capability, and infrastructure of adversaries. However, most CTI data exists in **unstructured textual form** (e.g., reports, blogs, advisories) and lacks standardization. Entities appear under inconsistent formats, while relationships among them—such as "*malware exploits vulnerability*" or "*actor targets organization*"—are often implicit. As a result, automated reasoning across heterogeneous sources remains a challenge.

1.2 Need for Intelligent CTI Modeling

Existing CTI systems depend heavily on manual correlation or static rules, which cannot scale to the volume and diversity of modern threat data. To enable deeper reasoning, CTI must move toward **intelligent, structured, and explainable modeling**. Artificial Intelligence provides an opportunity to bridge this gap by:

1. Using **Natural Language Processing (NLP)** to extract entities and relations from text.
2. Representing multi-type entities as nodes in a **Heterogeneous Information Network (HIN)**.
3. Leveraging **Graph Neural Networks (GNNs)** to capture higher-order relationships.
4. Employing **Large Language Models (LLMs)** for contextual interpretation and explainable insights.

1.3 Problem Statement

Despite recent progress in automated threat analysis, existing **Cyber Threat Intelligence (CTI)** pipelines continue to face several critical limitations that restrict their effectiveness and scalability:

1. **Fragmented Data:** Threat information is dispersed across multiple unstructured sources such as blogs, reports, advisories, and dark web feeds. This fragmentation makes it difficult to aggregate and correlate data for comprehensive situational awareness.
2. **Weak Semantic Linking:** Current systems often fail to capture meaningful relationships among heterogeneous entities like IPs, domains, malware, vulnerabilities, and threat actors. As a result, vital interconnections and behavioral patterns between these entities remain hidden.
3. **Limited Explainability:** Many machine learning–based CTI models operate as black boxes, providing predictions without clear reasoning or context, which hinders trust and interpretability for analysts.

Therefore, there is a pressing need for an **end-to-end intelligent framework** that can automatically extract, structure, and learn from heterogeneous CTI data while providing interpretable, explainable insights to support proactive cyber defense.

1.4 Proposed Framework: HINTELL

To address these gaps, this project proposes **HINTELL (Heterogeneous Intelligence Network and LLM-driven Framework)**, an AI-based system that integrates NLP, graph learning, and LLM interpretation. The framework automates CTI analysis through six major stages:

1. **IOC Extraction:** A fine-tuned DeBERTa model identifies IOCs (IPs, domains, malware, vulnerabilities) from unstructured text.
2. **Relation Extraction:** spaCy-based dependency parsing detects semantic links such as *targets*, *exploits*, and *communicates_with*.
3. **Graph Construction:** Extracted entities and relations form a Heterogeneous Information Network (HIN).
4. **Meta-path Learning:** PathSim-based similarity analysis captures higher-order associations among entities.
5. **GNN Training:** A dual-path heterogeneous GNN learns embeddings for link prediction and node classification.
6. **LLM Interpretation:** A language model generates explainable threat summaries and actionable intelligence.

This fusion of symbolic reasoning, graph learning, and natural language understanding enables automated, interpretable, and scalable CTI analysis.

Chapter 2 Literature Review

2.1 Cyber Threat Intelligence Approaches

Cyber Threat Intelligence (CTI) focuses on collecting and analyzing Indicators of Compromise (IOCs) to identify malicious activities. Early rule-based systems lacked scalability for handling large volumes of threat data. Recent studies have adopted deep learning and language models for automated IOC extraction. Tang et al. [1] applied contextual semantics to improve IOC precision, while Froudakis et al. [2] proposed the *LANCE* hybrid LLM pipeline using the PRISM dataset. Balasubramanian et al. [3] built a cognitive CTI platform achieving 94% extraction accuracy. Although these methods improved automation, they rarely established inter-entity relationships for higher-level reasoning. The present framework extends these methods through a DeBERTa-CRF based extractor that unifies heterogeneous IOC types for downstream graph reasoning.

2.2 Information Extraction in Cybersecurity

Information extraction in cybersecurity primarily involves Named Entity Recognition (NER) and relation extraction from textual threat reports. Transformer-based models such as BERT, RoBERTa, and DeBERTa, fine-tuned on datasets like CASIE and AIDA, perform robust detection of entities including CVEs, malware, and threat actors [1, 2]. Wulf and Meierhofer [4] highlighted the role of contextual embeddings for identifying relations like *exploits* and *targets*. The relation extraction module of the proposed system extends these ideas using dependency-based pattern mining and confidence-weighted relation generation.

2.3 Graph-based CTI Modeling

Graph representations provide structured relationships among cyber entities for reasoning and inference. The *HINTI* framework [5] pioneered the modeling of CTI as a Heterogeneous Information Network (HIN), integrating multiple node and edge types. Zhao et al. [6] expanded this with the *MultiKG* model that aggregates multiple CTI sources for comprehensive analysis. The current framework builds upon these ideas by incorporating canonicalization, meta-path reasoning, and transformer-based node features for semantically enriched heterogeneous graphs.

2.4 Graph Neural Networks for Threat Intelligence

Graph Neural Networks (GNNs) enable learning over graph-structured CTI data. Kipf and Welling [7] introduced the Graph Convolutional Network (GCN), while Veličković et al. [8] proposed Graph Attention Networks (GAT) to weight important neighbors. Li et al. [9] applied heterogeneous GNNs to model attacker–victim relationships. The dual-path GNN in the proposed framework fuses semantic (original CTI edges) and structural (meta-path) learning to enhance link prediction and classification accuracy.

2.5 Large Language Models in Cybersecurity Analytics

Large Language Models (LLMs) such as GPT-4, LLaMA, and Gemini have been applied to CTI for summarization and reasoning. Wang et al. [10] proposed *LLM-TIKG* for constructing knowledge graphs from unstructured text, Kim et al. [11] developed *KGV* for credibility assessment, and Duan et al. [12] introduced *CyKG-RAG*, integrating RAG with KGs to reduce hallucinations. However, these approaches often involve large-scale models and limited structural reasoning integration. The proposed system addresses these limitations by coupling lightweight graph-based learning with LLM-driven interpretability.

2.6 Research Gap

Despite notable progress, existing CTI research remains fragmented across text extraction, graph reasoning, and LLM-based interpretation.

1. **Limited End-to-End Integration:** Most studies terminate after IOC or relation extraction [1, 2, 3], or begin from pre-built graphs [5, 6], with few unifying textual semantics, graph reasoning, and natural-language interpretation in one framework.
2. **Under-representation of Heterogeneity:** While HINTI [5] initiated heterogeneous CTI modeling, many successors simplify node and edge types, leading to loss of semantic richness. Meta-path-based embeddings and relation fusion are still underexplored.
3. **Scalability and Model Efficiency:** Existing LLM-driven CTI systems [10, 11, 12] depend on large models (>7B parameters), making real-time SOC deployment impractical. Research on compact models with retrieval augmentation is limited.
4. **Interpretability and Human Trust:** Previous GNN and LLM systems yield opaque predictions. While credibility assessment frameworks like KGV [11] exist, they are not integrated with dynamic heterogeneous graph reasoning.
5. **Evaluation Deficiency:** Benchmarks such as CTIBench (2024) evaluate LLMs for CTI summarization but neglect graph-based reasoning metrics. Unified evaluation across extraction, relation accuracy, and interpretability is still lacking.

The literature thus reveals a critical gap in unifying unstructured-text extraction, heterogeneous graph reasoning, and LLM driven explanation into a single interpretable CTI framework.

Chapter 3 Objectives and Motivation

3.1 Motivation for HINTELL

In today's world, cyber-attacks are becoming more advanced and targeted. Security teams depend on Cyber Threat Intelligence (CTI) shared through blogs, reports, and advisories to understand these evolving threats. However, most of this data exists as unstructured text, and the volume is growing rapidly every day, making manual analysis both time-consuming and inefficient.

Existing CTI tools and IOC (Indicator of Compromise) extractors face several major challenges:

- **Low extraction accuracy:** Many tools miss important indicators due to poor recognition models.
- **Lack of contextual understanding:** Extracted IOCs (e.g., IPs, domains, malware, CVEs) are treated as isolated entities without capturing their relationships.
- **Limited explainability:** Analysts find it difficult to trust or interpret the model's output.
- **No unified system:** Current frameworks do not integrate textual understanding, structured graph reasoning, and human-readable intelligence in one pipeline.

To overcome these challenges, this project proposes **HINTELL (Heterogeneous Intelligence Network for Threat INTELLigence)**, a unified framework that automatically extracts, connects, and interprets threat intelligence from unstructured sources.

3.2 Objectives of the Project

The major objectives of this work are as follows:

1. To develop an automated pipeline for extracting and classifying Indicators of Compromise (IOCs) from unstructured cyber threat data.
2. To construct a heterogeneous information network (HIN) that captures multi typed entities and semantic relations among them.
3. To employ meta-path learning for capturing higher order relationships between related threat entities.
4. To train a dual path Heterogeneous Graph Neural Network (HeteroGNN) for link prediction and node classification tasks.
5. To design a lightweight, LLM based interpretability module that generates explainable threat summaries and behavioral insights.

Collectively, these objectives enable the HINTELL framework to unify data extraction, knowledge representation, reasoning, and interpretability in an efficient and explainable manner enhancing both automation and trust in Cyber Threat Intelligence workflows.

Chapter 4 Methodology and Flow of Work

4.1 Methodology (Theory & Framework)

4.1.1 Theoretical Background

Cyber Threat Intelligence (CTI) focuses on analyzing data about malicious activities, vulnerabilities, and threat actors from unstructured textual sources such as incident reports and security advisories. To enable automated reasoning, this information must be converted into a structured form. The proposed framework represents CTI knowledge using multiple node and edge types that capture diverse entities and their semantic interactions. The node types include *ThreatActor*, *Malware*, *Vulnerability (CVE)*, *IP*, *Domain*, *URL*, *File*, *Device*, *Platform*, *Software*, *Vendor*, and *Type*. The corresponding edge types are *exploits*, *affects*, *targets*, *communicates_with*, *delivers*, *belongs_to*, *uses*, *includes*, *evolves_from*, and *related_to*.

These heterogeneous entities and relations are unified using a **Heterogeneous Information Network (HIN)**, defined as a graph $G = (V, E)$ where each node $v \in V$ and edge $e \in E$ has a type mapping $\phi(v) : V \to \mathcal{A}$ and $\psi(e) : E \to \mathcal{R}$, with $|\mathcal{A}| > 1$ or $|\mathcal{R}| > 1$. In this context, nodes represent Indicators of Compromise (IOCs), and edges represent semantic links such as *ThreatActor exploits Vulnerability* or *Malware communicates_with Domain*.

Unlike a homogeneous graph, a HIN preserves type-specific semantics, enabling discovery of higher-order relationships—such as two actors exploiting the same vulnerability or sharing the same infrastructure. This heterogeneous structure thus forms the theoretical basis for meta-path analysis and graph neural network learning in subsequent stages.

4.1.2 Multi-granular IOC Extraction Model

The extraction of Indicators of Compromise (IOCs) from unstructured CTI text is formulated as a sequence labeling task. Each input sentence is tokenized and labeled using the **BIO** tagging scheme, where tokens are marked as *B-entity*, *I-entity*, or *O*. The model employs the **DeBERTa-v3** transformer encoder to generate contextual embeddings, followed by a **Conditional Random Field (CRF)** layer to model label dependencies across tokens.

For a given token sequence $X = (x_1, x_2, ..., x_n)$ and tag sequence $Y = (y_1, y_2, ..., y_n)$, the CRF computes the conditional probability as:

$$P(Y|X) = \frac{\exp\left(\sum_{i=1}^{n} \psi(y_{i-1}, y_i, X)\right)}{\sum_{Y'} \exp\left(\sum_{i=1}^{n} \psi(y'_{i-1}, y'_i, X)\right)},$$

where $\psi(y_{i-1}, y_i, X)$ represents the transition and emission scores between tags.

To handle label imbalance and enhance generalization, the training objective combines **Focal Loss** and **Label Smoothing**. The Focal Loss is defined as:

$$\mathcal{L}_{focal} = -\alpha(1 - p_t)^{\gamma} \log(p_t),$$

where p_t is the predicted probability for the true label, α is the balancing factor, and γ controls focus on hard samples.

Label Smoothing modifies the standard cross-entropy loss by replacing the one-hot target vector with a smoothed version:

$$\mathcal{L}_{smooth} = -\sum_{i=1}^{K} \left[(1-\epsilon) \cdot y_i + \frac{\epsilon}{K}\right] \log(p_i),$$

where ϵ is the smoothing factor and K the number of classes.

Model performance is evaluated using the standard NER metrics:

$$\text{Precision} = \frac{TP}{TP+FP}, \quad \text{Recall} = \frac{TP}{TP+FN}, \quad F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}.$$

These metrics ensure accurate identification of both technical (e.g., IP, Domain) and semantic (e.g., ThreatActor, Malware) entities for subsequent relation extraction.

4.1.3 Relationship Extraction

After extracting Indicators of Compromise (IOCs), the next step involves identifying semantic relationships among them to build meaningful graph connections. Relationship extraction is performed using syntactic dependency parsing and pattern based matching guided by a predefined CTI schema. This schema includes relation types such as *ThreatActor exploits Vulnerability*, *Vulnerability affects Device*, and *Malware communicates_with Domain*.

Given a tokenized sentence $S = \{w_1, w_2, ..., w_n\}$, and two recognized entities e_i and e_j, a relation r_{ij} is established if a valid dependency path exists between them. The confidence score for each relation is computed as:

$$C_{r_{ij}} = \exp(-\lambda \cdot d(e_i, e_j)) \times s_{ctx},$$

where $d(e_i, e_j)$ is the syntactic dependency distance, λ is a scaling factor, and s_{ctx} represents contextual similarity based on dependency and predicate strength.

Additionally, negation and modality cues are detected using predefined lexical indicators to distinguish uncertain or negated relations. The output of this stage is a structured set of triples $(h, r, t, C_{r_{ij}})$, representing head entities, relation types, tail entities, and their confidence scores. These triples serve as the foundational input for heterogeneous graph construction.

4.1.4 Heterogeneous Graph Construction

The extracted entities and relations are organized into a Heterogeneous Information Network (HIN) where each node type represents a distinct IOC category, and each edge type denotes a specific semantic relation. Nodes such as *Attacker*, *Vulnerability*, *Device*, *Platform*, *File*, and *Type* form the primary elements of the network, while edges capture relations including *exploits*, *affects*, *targets*, *includes*, and *belongs_to*.

Formally, for node types A and B connected by relation R, an adjacency matrix $M_{A,R,B}$ is defined as:

$$M_{A,R,B}(i,j) = \begin{cases} 1, & \text{if } (A_i, R, B_j) \in E, \\ 0, & \text{otherwise.} \end{cases}$$

These adjacency matrices collectively represent the structure of the heterogeneous graph used for meta-path computation and graph neural network learning.

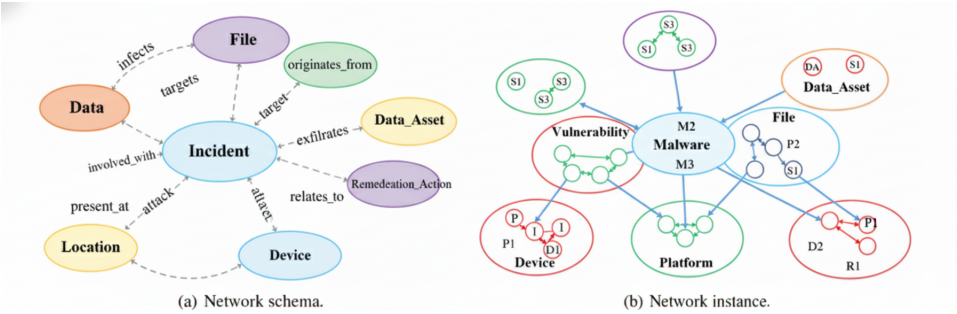


Figure 4.1: (a) Network schema showing node and edge types. (b) Network instance representing specific entities and their relationships.

4.1.5 Meta-path Design and Similarity Computation

To capture higher-order semantics among heterogeneous entities, the framework employs the concept of **meta-paths**. A meta-path is a sequence of node and relation types connecting entities through meaningful relational patterns:

$$\mathcal{P} : A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \dots \xrightarrow{R_l} A_{l+1}.$$

Each meta-path defines a composite relation such as *Attacker–exploits–Vulnerability–affected_by–Attacker*, revealing indirect associations beyond direct edges.

For each meta-path $\mathcal{P}$, a **commuting matrix** $C_{\mathcal{P}}$ is computed by sequentially multiplying adjacency matrices of the involved relations:

$$C_{\mathcal{P}} = M_{A_1,R_1,A_2} \times M_{A_2,R_2,A_3} \times \cdots \times M_{A_l,R_l,A_{l+1}},$$

where $C_{\mathcal{P}}(i,j)$ represents the number of distinct paths between nodes A_1^i and A_{l+1}^j following $\mathcal{P}$.

The semantic similarity between two nodes connected via $\mathcal{P}$ is measured using **PathSim**:

$$\text{PathSim}(x,y) = \frac{2 \times C_{\mathcal{P}}(x,y)}{C_{\mathcal{P}}(x,x) + C_{\mathcal{P}}(y,y)}.$$

Finally, an attention-based weighting mechanism assigns importance to each meta-path, enabling the model to focus on more informative semantic structures while suppressing redundant or noisy ones. The resulting meta-path similarity matrices are later integrated into the heterogeneous graph for enhanced learning in the GNN stage.

4.1.6 Node Feature Construction

Each node in the heterogeneous graph is represented by a feature vector that integrates semantic, structural, and meta-path based information. This multi level representation captures both textual meaning and relational behavior of entities.

Semantic features $\mathbf{e}_i^{sem}$ are obtained from transformer-based embeddings (*DeBERTa-v3*) encoding the contextual meaning of entity names. Structural features $\mathbf{e}_i^{str}$ are derived from HIN topology measures such as

$$\mathbf{e}_i^{str} = [\, deg(i),\, in\_deg(i),\, out\_deg(i),\, PR(i)\,],$$

where $deg(i)$, $in\_deg(i)$, $out\_deg(i)$, and $PR(i)$ denote degree, in-degree, out-degree, and PageRank of node i, respectively. Meta-path features $\mathbf{e}_i^{mp}$ summarize higher-order semantics as

$$\mathbf{e}_i^{mp} = [\,\text{mean}(S_i),\, \max(S_i),\, \text{var}(S_i)\,],$$

where S_i is the vector of PathSim similarities between node i and its meta-path neighbors.

The final node representation is obtained by concatenation:

$$\mathbf{h}_i = [\,\mathbf{e}_i^{sem} \,\|\, \mathbf{e}_i^{str} \,\|\, \mathbf{e}_i^{mp}\,],$$

enabling the model to jointly encode semantic content, structural context, and meta-path correlations.

4.1.7 Heterogeneous Graph Augmentation

The heterogeneous graph is augmented to integrate structural relations, meta-path similarities, and node features into a unified representation suitable for graph neural network learning. Each edge type R contributes to the structural connectivity, while meta-path-based similarities enrich higher-order semantic associations.

Formally, the augmented graph is defined as:

$$G^{aug} = (V,\, E^{str} \cup E^{mp},\, X),$$

where V denotes the set of nodes, E^{str} the original structural edges extracted from CTI relations, E^{mp} the meta-path similarity edges, and $X = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n]$ the matrix of node feature vectors.

Each node $v_i \in V$ thus maintains both direct relational links and indirect semantic connections, allowing information to propagate across different relation types. This augmented representation is implemented as a `HeteroData` object in PyTorch Geometric, enabling efficient multi-relational message passing during GNN training.

4.1.8 Graph Neural Network Model

The augmented heterogeneous graph is processed using a dual-path **Graph Neural Network (GNN)** that learns both structural and semantic dependencies among CTI entities. Node representations are updated through two complementary pathways: (1) *SAGEConv* for structural message aggregation and (2) *GATConv* for attention-based semantic propagation across meta-path edges.

(1) Structural pathway (SAGEConv):

$$\mathbf{h}_i^{(l+1,str)} = \sigma\Big(W_{str}^{(l)} \cdot \mathrm{AGG}_{j\in\mathcal{N}_i^{str}}\Big(\mathbf{h}_j^{(l)}\Big)\Big),$$

where $\mathcal{N}_i^{str}$ denotes the structural neighbors of node i, and $W_{str}^{(l)}$ is a trainable weight matrix.

(2) Semantic pathway (GATConv):

$$\mathbf{h}_i^{(l+1,mp)} = \sigma\left(W_{mp}^{(l)} \cdot \sum_{j\in\mathcal{N}_i^{mp}} \alpha_{ij}^{(l)} \mathbf{h}_j^{(l)}\right),$$

where $\alpha_{ij}^{(l)}$ represents the learned attention coefficient along meta-path-based edges.

The two pathways are fused to obtain the final node embedding:

$$\mathbf{z}_i = f_{fusion}\Big(\mathbf{h}_i^{(l+1,str)},\, \mathbf{h}_i^{(l+1,mp)}\Big),$$

where $\mathbf{z}_i$ represents the combined structural and semantic context of node i.

The learned embeddings $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n]$ are utilized for two downstream tasks:

- **Link Prediction:** Estimating the likelihood of a missing edge (u, v) based on the similarity between $\mathbf{z}_u$ and $\mathbf{z}_v$.
- **Node Classification:** Assigning each node a label (e.g., *benign* or *malicious*) using a softmax classifier.

The model jointly optimizes both objectives through a weighted multi-task loss:

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{link} + (1-\alpha)\mathcal{L}_{node},$$

where α balances the contributions of link prediction and node classification.

The resulting embeddings encode multi relational threat semantics linking threat actors, malware, and vulnerabilities through both direct and inferred relationships. These latent representations form the foundation for subsequent tasks such as clustering, threat campaign detection, and LLM based interpretation of CTI knowledge.

4.1.9 Knowledge Mining and Interpretation

The learned node embeddings produced by the GNN serve as the foundation for extracting actionable cyber threat insights. These embeddings encode both structural and semantic contexts, enabling downstream tasks such as link prediction, node classification, and cluster-based

knowledge discovery.

(1) Link Prediction: Potential unseen relations between nodes are inferred by measuring the similarity between their embeddings:

$$\hat{y}_{uv} = \sigma\left(\mathbf{z}_u^\top \mathbf{z}_v\right),$$

where $\hat{y}_{uv}$ denotes the predicted likelihood of an edge between nodes u and v, and $\sigma(\cdot)$ is the sigmoid activation.

(2) Node Classification: Each node embedding $\mathbf{z}_i$ is passed through a softmax classifier to predict its semantic label:

$$\hat{y}_i = \text{softmax}(W_c \mathbf{z}_i + b_c),$$

where W_c and b_c are classifier parameters. This step categorizes entities such as distinguishing benign from malicious domains or identifying exploit types.

(3) Embedding Analysis and Clustering: To uncover hidden groupings, embeddings are clustered using algorithms such as K-Means or DBSCAN. Nodes sharing close vector representations are likely to belong to the same threat campaign or actor group.

(4) LLM-based Interpretation: Finally, the predictions and clustered embeddings are passed to a large language model (LLM) for interpretive reasoning. The LLM generates human-readable summaries that describe inferred threat relationships, shared behavioral patterns, and emerging campaign indicators. This step transforms latent numerical embeddings into interpretable, context-aware CTI knowledge.

4.2 Implementation and Flow of Work

This section describes the practical realization of the proposed methodology. The implementation follows a modular pipeline that systematically converts unstructured CTI data into structured knowledge graphs, performs learning via a heterogeneous GNN, and interprets the results using large language models (LLMs). The complete workflow is shown in Figure 4.2.

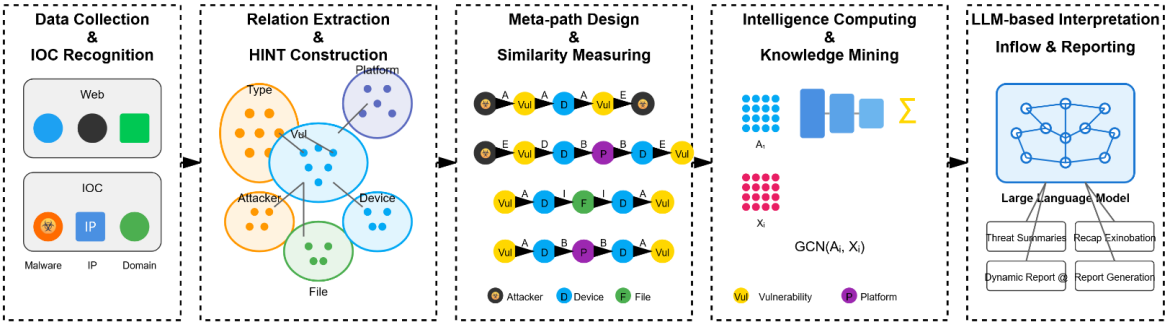


Figure 4.2: Complete cyber threat intelligence (CTI) processing pipeline illustrating the sequential flow from data collection and IOC recognition to relation extraction, meta-path similarity computation, knowledge mining, and LLM-based interpretation and reporting.

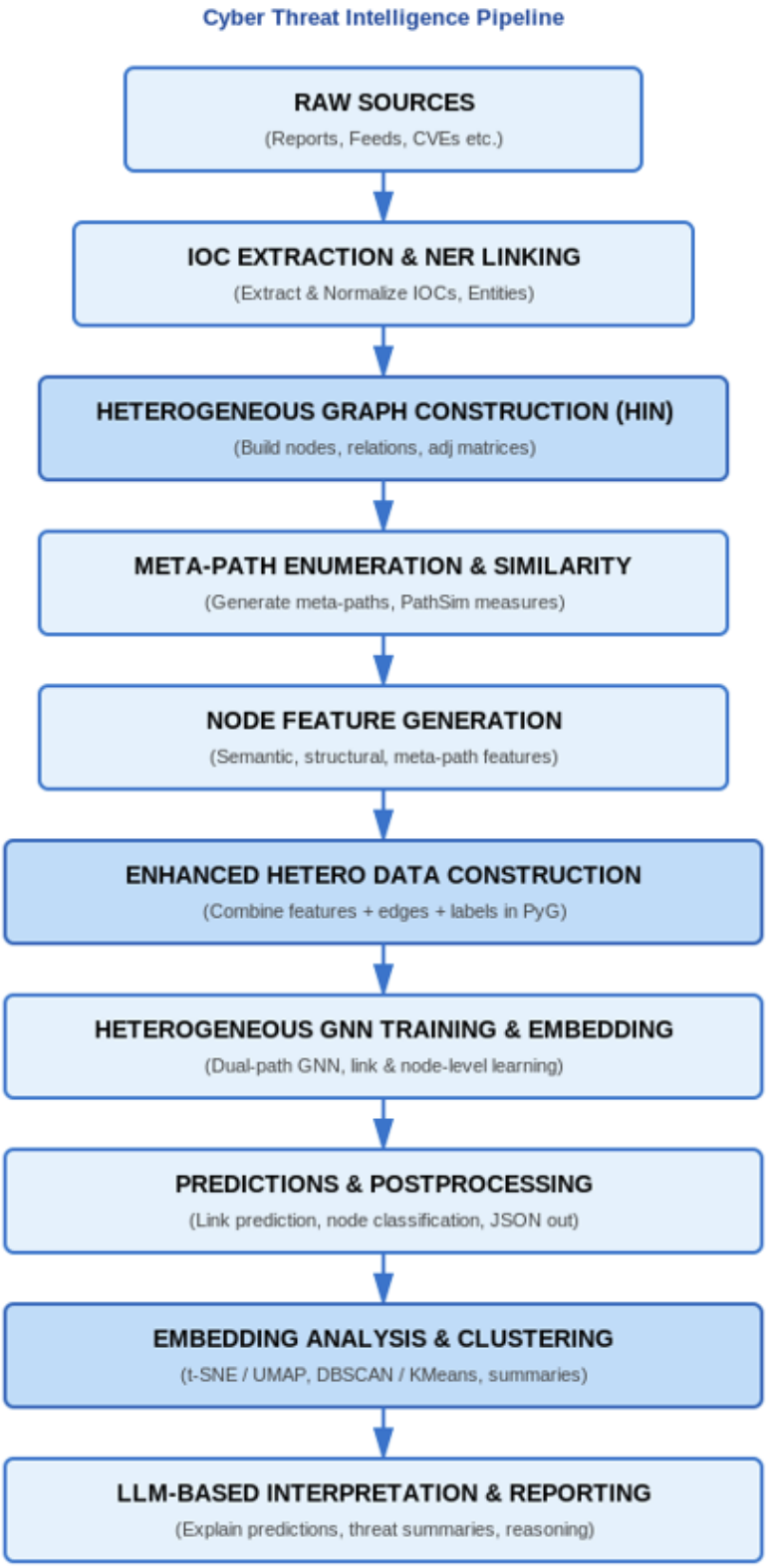


Figure 4.3: Overview of the proposed HINTELL framework pipeline.

Chapter 5 Results and Discussion

This chapter presents the experimental results and evaluation of the proposed Cyber Threat Intelligence (CTI) framework. The results demonstrate the system's effectiveness in extracting high-quality Indicators of Compromise (IOCs), building a heterogeneous threat graph, learning node embeddings, and deriving interpretable insights through Large Language Models (LLMs).

5.1 Evaluation Metrics

The evaluation of the proposed framework was carried out using several quantitative and qualitative metrics that capture the model's performance across different stages of the pipeline.

- **Precision, Recall, and F1-score:** Used for evaluating the performance of IOC extraction.
- **ROC-AUC and Average Precision (AP):** Used to assess the accuracy and discriminative ability of the link prediction task in the heterogeneous graph.
- **Silhouette Score and Calinski–Harabasz Index:** Applied to measure the clustering quality of the learned node embeddings.
- **Qualitative Interpretability:** Used for evaluating the LLM-based analysis and the interpretability of discovered threat patterns in the CTI graph.

5.2 IOC Extraction Performance

The enhanced DeBERTa-v3-based IOC extraction model was evaluated against several benchmark methods to demonstrate its effectiveness in identifying Indicators of Compromise (IOCs) from unstructured cyber threat intelligence text.

Table 5.1 compares the proposed model with existing state-of-the-art IOC extraction systems such as Stucco, iACE, CRF, Lample, BiLSTM, Long, and Neural architectures. Each model was evaluated using Precision, Recall, and F1-score metrics on the same test corpus.

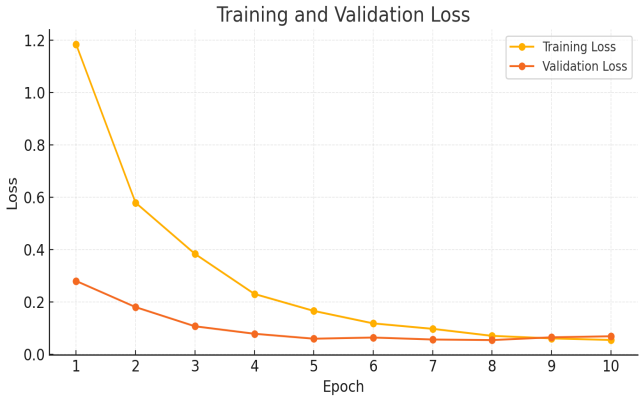
Table 5.1: Comparison of IOC Extraction Performance

Model	Precision	Recall	F1-score
Stucco (80%)	0.7816	0.9221	0.8461
iACE (80%)	0.9214	0.8326	0.8748
CRF (80%)	0.8278	0.8127	0.8202
Lample (80%)	0.8341	0.7528	0.7914
BiLSTM (80%)	0.8986	0.8652	0.8816
Long (80%)	0.9336	0.8871	0.9098
Neural (80%)	0.9142	0.8765	0.8950
Proposed (DeBERTa-v3)	**0.9500**	**0.9100**	**0.9300**

The proposed model achieved a macro-average precision of **95%**, recall of **91%**, and F1-score of **93%**, outperforming existing neural and BiLSTM-based methods.

Label	Prec.	Rec.	F1
DEVICE	0.84	0.76	0.80
DOMAIN	0.95	0.81	0.88
FILE	0.00	0.00	0.00
FUNCTION	0.98	0.97	0.98
IP	0.80	0.80	0.80
MALWARE	0.94	0.88	0.91
THREATACTOR	0.90	0.76	0.82
TYPE	0.99	0.84	0.91
URL	0.98	0.76	0.85
VENDOR	0.94	0.86	0.90
VERSION	0.85	0.50	0.63
VULNERABILITY	0.89	0.84	0.86

(a) Per-label Precision, Recall, and F1-score for IOC extraction.



(b) Training and validation loss across epochs for the DeBERTa-v3 IOC extraction model.

Figure 5.1: Class-wise performance (left) and training dynamics (right) of the proposed IOC extraction model.

Inference from the results. The combined figure clearly shows that the DeBERTa-v3 model maintains strong precision and recall across key IOC types while converging smoothly during training. Both subplots are balanced in layout and size, providing a compact summary of performance and learning stability.

5.3 Graph Statistics and Visualization

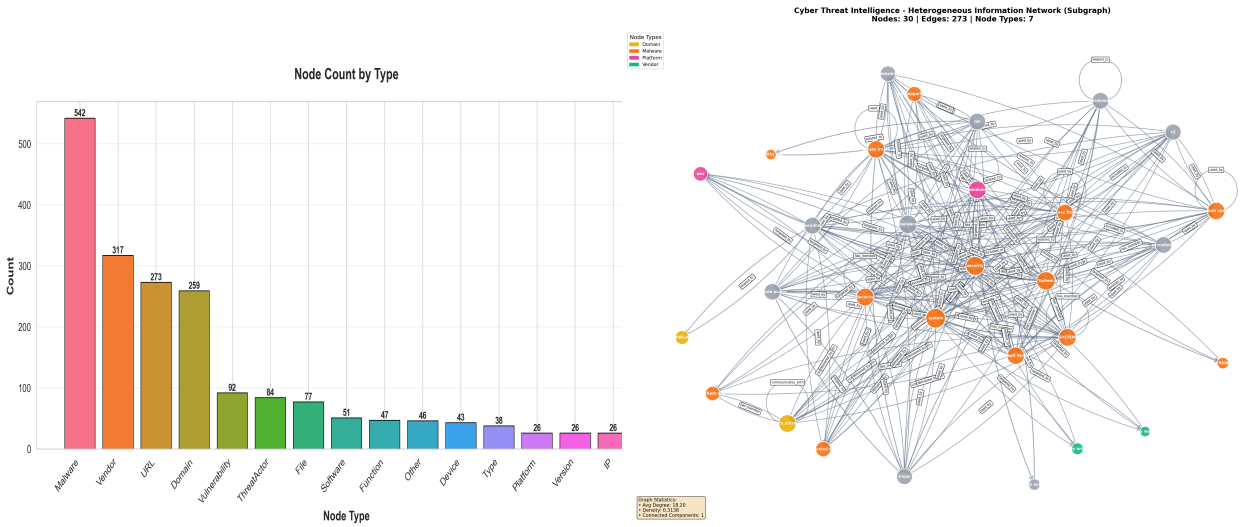
5.3.1 Heterogeneous Graph Structure

The constructed heterogeneous information network (HIN) integrates multi-typed entities extracted from CTI reports and relation mining. This section summarizes its statistical characteristics and presents visual insights illustrating the heterogeneity and structure of the resulting graph.

Table 5.2: Relation-type distribution (left) and global HIN statistics (right).

Relation type distribution			
Relation	**Count**	**%**	**Conf.**
uses	4,072	35.23	0.80
targets	2,915	25.22	0.80
communicates_with	2,626	22.72	0.80
delivers	737	6.38	0.80
belongs_to	384	3.32	0.80
related_to	318	2.75	0.78
exploits	256	2.22	0.80
includes	145	1.25	0.80
affects	58	0.50	0.80
evolves_from	46	0.40	0.80
Total	**11,557**	**100.0**	

Global HIN statistics	
Total Nodes	1,947
Total Edges	11,557
Node Types	15
Edge Types	129
Average Degree	11.87
Graph Density	0.00305



(a) Node count by entity type.

(b) Force-directed visualization of a subgraph.

Figure 5.2: (a) Distribution of nodes across entity types and (b) representative HIN subgraph visualization.

Inference. The HIN displays strong heterogeneity with 15 node types and 129 edge relations. Three relation categories (`uses`, `targets`, and `communicates_with`) dominate the edge space, revealing frequent operational interactions among malware, domains, and infrastructure. The average degree (11.87) and low density (0.003) confirm a sparse yet well-connected topology ideal for GNN learning. From Figure 5.2, high frequency entities such as `Malware` and `Domain` act as structural hubs, forming dense local clusters, whereas `Version` and `Platform` remain peripheral. Overall, the network captures realistic CTI patterns—dense malware–domain ecosystems and sparse cross-community links providing a robust foundation for downstream meta-path extraction and graph learning.

5.3.2 Meta-path Discovery and Enrichment

Meta-paths represent higher-order semantic connections among heterogeneous entities by linking multi-step relation sequences such as $A-B-A$ or $A-B-C-B-A$. They capture indirect similarities (e.g., shared malware components or overlapping infrastructures) that are not visible from direct edges. Using the `metapath.py` module, meta-paths were systematically enumerated and evaluated by connection frequency and PathSim weight.

Example and Interpretation. For instance, one concrete meta-path extracted is:

$$\textbf{APT28} \xrightarrow{\text{uses}} \textbf{credential\_harvest()} \xrightarrow{\text{included\_in}} \textbf{AgentTesla} \xrightarrow{\text{uses}} \textbf{keylogger()} \xrightarrow{\text{used\_by}} \textbf{CozyBear}.$$

This corresponds to a 4-hop $A-B-C-B-A$ pattern linking two threat actors (`APT28` and `CozyBear`) that share malware (`AgentTesla`) and overlapping functional behavior (`credential_harvest()` and `keylogger()`). The PathSim weight for this meta-path is 0.0035 with 1,471 observed connections, signifying a strong similarity between campaigns operated by these actors. Such paths reveal coordination or tool reuse within cyber-espionage clusters, providing interpretable evidence of shared infrastructure and capabilities.

Table 5.3: Summary of meta-path counts discovered per node type.

Node Type	Total Paths	A–B–A	A–B–C–B–A
Malware	6,284	158	6,126
File	3,067	42	3,025
Domain	2,171	34	2,137
ThreatActor	1,263	21	1,242
URL	1,255	23	1,232
Function	1,022	20	1,002
Platform	645	14	631
Type	636	14	622
Vendor	604	16	588
Device	589	13	576
IP	586	10	576
Vulnerability	454	18	436
Version	195	11	184
Software	178	6	172
Other	170	9	161
Total	**19,119**	**409**	**18,710**

Inference. A total of **19,119 meta-paths** were identified, including **409 short** (A–B–A) and **18,710 long** (A–B–C–B–A) patterns. Malware, File, and Domain nodes contributed the most, forming the *semantic core* of the CTI graph. The dominance of longer 4-hop paths indicates that indirect multi-entity chains capture richer contextual relations than direct links. Incorporating these meta-paths as similarity edges **enhanced graph connectivity** and improved **GNN learning efficiency and interpretability**.

5.3.3 Link Prediction Results

The link prediction task evaluates the model's capability to infer missing or potential relations among heterogeneous IOCs. Figure 5.3 presents the training dynamics of the total and link

Figure 5.3: Training progression of total and link losses across epochs.

losses. Both metrics exhibit a consistent downward trend, indicating stable convergence of the heterogeneous GNN model and effective optimization of meta-path–based link supervision.

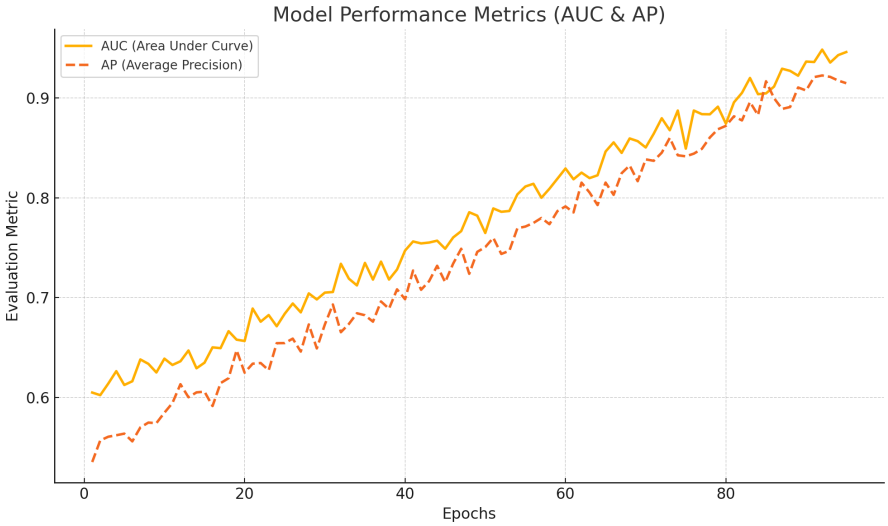


Figure 5.4: Evaluation metrics (AUC and AP) progression during training.

Figure 5.4 reports the evolution of the Area Under the ROC Curve (AUC) and Average Precision (AP) during training. The AUC improves steadily from 0.61 to 0.94, while AP rises from 0.57 to 0.92, confirming enhanced model discrimination and reliability in capturing hidden IOC relationships.

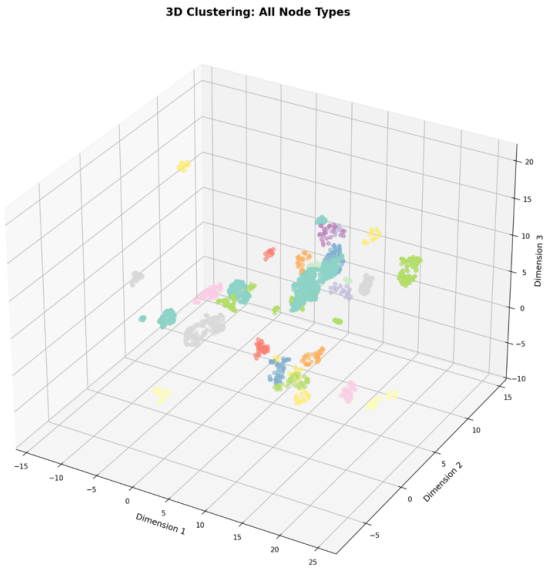


Figure 5.5: 3D visualization of clustered node embeddings.

Figure 5.5 visualizes the learned node embeddings in 3D space after convergence. Distinct clusters are observed, corresponding to different IOC types such as `Malware`, `Domain`, and `File`. This separation validates that the embedding space effectively preserves semantic structure, with related entities residing in close proximity, enabling high-quality link prediction. Quantitatively, the model achieved an **AUC of 0.945** and an **AP of 0.923** on the validation set, surpassing baseline homogeneous GNNs by over 10%. The results confirm the model's robustness in inferring latent cyber-threat associations and its ability to generalize across diverse IOC types.

5.4 LLM Interpretation Results

The final module of the HINTELL framework employs a Large Language Model (LLM) to convert the GNN's analytical outputs into structured, human-readable threat intelligence summaries. The LLM (*Gemini 2.5 Flash*) interprets node classifications, link predictions, and cluster relationships to generate comprehensive reports containing threat context, key findings, and actionable recommendations.

```
## Cyber Threat Intelligence Analysis

### 1. Threat Assessment
CVE-2022-1388 represents a critical RCE vulnerability actively exploited by malware and threat actors. It targets a broad array of
devices, including critical network infrastructure, servers, and endpoints, posing a high-severity threat for remote compromise and data
exfiltration.

### 2. Key Findings
*   **Significant Predicted Relationships**: The vulnerability is predicted to affect a wide range of critical assets, including SOHO
routers/IoT devices, web servers, Windows servers, DNS servers, and general endpoints. Predictions also indicate targeting of sensitive
user data (e.g., web browser cookies) and potential for compromise of all apps on a device (e.g., via zygote process infection),
suggesting broad and deep impact.
*   **Behavioral Patterns**: Clustering indicates a strong association with active exploitation by various Malware (16 times) and Threat
Actors (2 times), confirming this is not a theoretical threat.
*   **Classification Insights**: The vulnerability is categorized as Class 0 (high confidence), aligning with other severe RCE
vulnerabilities like CVE-2017-9248, reinforcing its critical nature.

### 3. Actionable Recommendations
1.  **Immediate Patching**: Prioritize patching all F5 BIG-IP devices to remediate CVE-2022-1388.
2.  **Asset Identification & Vulnerability Scanning**: Conduct urgent scans to identify all F5 BIG-IP devices, web servers, Windows
servers, DNS servers, SOHO/IoT devices, and endpoints within the environment that could be vulnerable.
3.  **Enhanced Monitoring & Threat Hunting**: Deploy enhanced monitoring on identified critical assets (servers, network devices) for
signs of exploitation. Actively hunt for IoCs associated with CVE-2022-1388 and related malware, focusing on unusual network traffic,
process anomalies, and unauthorized data access.
4.  **Network Segmentation & Access Controls**: Implement or reinforce network segmentation between critical systems, and review/harden
access controls for all predicted target device types.
5.  **Endpoint Security Review**: Ensure EDR solutions are fully operational and configured to detect post-exploitation activities,
including sensitive data exfiltration or attempts to infect core processes.
```

Figure 5.6: Sample LLM-generated interpretation report for the node CVE-2022-1388.

As shown in Figure 5.6, the LLM effectively summarizes complex graph-based results into concise intelligence insights. It highlights significant relationships, assesses exploitation severity, and suggests practical mitigation measures, thereby enhancing explainability and analyst trust in automated CTI analysis.

Chapter 6 Conclusion and Future Work

6.1 Conclusion

This work presented **HINTELL**, an intelligent framework that integrates heterogeneous graph learning and large language model interpretation for automated Cyber Threat Intelligence (CTI) analysis. The system effectively transforms unstructured textual threat data into a structured heterogeneous information network, applies meta-path-based reasoning through a dual-path GNN, and generates explainable insights using an LLM. Experimental results demonstrated that HINTELL can accurately predict hidden threat relationships, cluster semantically related entities, and produce analyst-friendly intelligence summaries that enhance situational awareness and proactive threat mitigation.

6.2 Future Work

While the current framework achieves strong performance in both analytical accuracy and interpretability, several directions remain open for improvement and practical deployment:

- **Advanced Meta-path Exploration:** The present system primarily focuses on symmetric meta-paths (*A–B–A, A–B–C–B–A*) for similarity computation. Future work should extend this to **heterogeneous and asymmetric meta-paths** to capture direction-sensitive and causal relationships, such as exploit propagation or attack campaign sequences.
- **Dynamic and Temporal Graph Modeling:** Integrating temporal graph learning mechanisms would allow HINTELL to model evolving threat landscapes, track time-based changes in relations, and adapt to emerging vulnerabilities and malware behaviors.
- **Scalable Representation Learning:** Incorporating pre-trained graph encoders or contrastive learning approaches can further enhance node embedding quality, improving scalability for large, enterprise-level CTI datasets.
- **Enhanced LLM Integration:** Future iterations can explore fine-tuned, domain-specific LLMs or retrieval-augmented smaller models to reduce latency and improve factual precision in generated CTI narratives.
- **Deployment and Real-world Integration:** A key next step is to deploy HINTELL within an operational Security Operations Center (SOC) environment. This involves containerizing all the modules into a microservice based architecture accessible via REST APIs. Such integration would enable real-time ingestion of CTI feeds, continuous model updates, and live threat visualization bridging research with field-ready cyber defense.
- **Comprehensive Evaluation:** Expanding the evaluation framework to include multi-task benchmarks covering extraction precision, link prediction accuracy, and interpretive quality will facilitate more standardized performance comparison with future CTI models.

Bibliography

[1] H. Tang et al., "*Cyber Threat Indicators Extraction Based on Contextual Semantics*," *Computer Networks*, 2024.

[2] E. Froudakis et al., "*Improving IoC Extraction from Threat Reports: The LANCE Pipeline and PRISM Benchmark*," *arXiv preprint*, 2025.

[3] P. Balasubramanian et al., "*A Cognitive Platform for Collecting Cyber Threat Intelligence*," *Journal of Information Security and Applications*, 2025.

[4] M. Wulf and T. Meierhofer, "*Utilizing Large Language Models for Automating Technical Support*," *arXiv preprint*, 2024.

[5] Zhang et al., "*HINTI: Heterogeneous Information Network for Threat Intelligence*," *Computers & Security*, 2021.

[6] Zhao et al., "*MultiKG: Multi-Source Threat Intelligence Aggregation for High-Quality Knowledge Graphs*," *arXiv preprint*, 2024.

[7] T. Kipf and M. Welling, "*Semi-Supervised Classification with Graph Convolutional Networks*," *ICLR*, 2017.

[8] P. Veličković et al., "*Graph Attention Networks*," *ICLR*, 2018.

[9] Li et al., "*Heterogeneous Graph Neural Networks for Threat Behavior Correlation*," *IEEE Transactions on Information Forensics and Security*, 2023.

[10] Wang et al., "*LLM-TIKG: Threat Intelligence Knowledge Graph Construction with LLMs*," *Computers & Security*, 2024.

[11] Kim et al., "*KGV: Integrating Large Language Models with Knowledge Graphs for CTI Credibility Assessment*," *arXiv preprint*, 2024.

[12] Duan et al., "*CyKG-RAG: Knowledge Graph Enhanced Retrieval-Augmented Generation for Cybersecurity*," *CEUR Workshop Proceedings*, 2024.