MEHGB: Meta-Path Pruned and Explainable Heterogeneous Graph Backdoor for Stealthy NIDS Evasion

Zhonghang Sui, Fei Kang*, Yuntian Zhao, Yan Guang
Key Laboratory of Cyberspace Security, Ministry of Education, China
Information Engineering University, Zhengzhou 450001, China
harry_hang@foxmail.com, kfminnie@163.com, yuntianzhao1105@163.com, gyinarmy@126.com

***Abstract*—Heterogeneous Graph Neural Networks (HGNNs) have become essential for encrypted traffic analysis, yet existing backdoor attacks fail to exploit their vulnerabilities due to a fundamental mismatch with homogeneous assumptions. Current methods rely on additive assumptions that fail to capture cooperative interactions for precise localization, suffer from signal dilution caused by massive structural redundancy, and generate semantically invalid triggers that violate rigid network protocols.**

To bridge this gap, we propose MEHGB (*M*eta-Path Pruned and *E*xplainable *H*eterogeneous *G*raph *B*ackdoor), a lightweight framework designed to expose the vulnerabilities of NIDS. The method uniquely integrates meta-path guided pruning to distill the informative graph core, a Conditional Random Field (CRF) explainer to capture non-linear cooperative interactions for precise localization, and constraint-driven optimization to enforce strict structural and statistical compliance. Empirical evaluations on UNSW-NB15, CSE-CIC-IDS2018, and CTU-Mixed-Capture demonstrate that MEHGB outperforms state-of-the-art baselines by 9.6% in Attack Success Rate while reducing trigger generation latency by 53.6%. Furthermore, the framework exhibits strong task-agnostic transferability and resilience against advanced defenses. Ultimately, by exposing these latent vulnerabilities, the study serves as a critical stress test to guide the development of next-generation NIDS architectures resilient to complex heterogeneous threats.

***Index Terms*—Heterogeneous Graph Neural Network, Backdoor Attack, Network Intrusion Detection, Graph Representation Learning, Explainable AI, Security and Privacy.**

I. Introduction

The widespread adoption of end-to-end encryption protocols, such as TLS 1.3 and QUIC, has rendered traditional payload-based Network Intrusion Detection Systems (NIDS) ineffective [1]. Consequently, recent research has pivoted to Heterogeneous Graph Neural Networks (HGNNs) which model encrypted traffic as structured graphs of flows, hosts, and services [2]. While these models capture complex latent correlations, applying existing graph backdoors to this domain presents a fundamental mismatch. Most prior attacks are tailored for generic or homogeneous graphs and fail to navigate the high-dimensional heterogeneity and massive structural redundancy of traffic [3], [4]. They typically rely on heuristic perturbations that ignore non-linear cooperative interactions inherent in HGNN message passing [5], leading to signal dilution and fragmented trigger placement. More critically, the neglect of strict network protocol constraints results in semantically invalid triggers, such as out-of-bound features, which are easily rejected by protocol parsers or consistency checks [6].

These limitations motivate the need to bridge the gap between graph adversarial learning and the operational reality of NIDS. Specifically, three critical challenges are identified: **Challenge 1: Reliable Trigger Localization.** The first hurdle is determining *where* to implant triggers for maximum stealth and impact. Conventional attribution methods (e.g., GNNExplainer [7]) rely on additive decomposition assumptions, treating node contributions as independent [8]. However, recent findings demonstrate that message passing in GNNs relies heavily on structural dependencies, where the influence of a node is cooperative and conditioned on its connected neighbors rather than isolated [5], [9]. Existing attacks fail to model these conditional correlations, leading to fragmented trigger placement that ignores the synergistic effects of local substructures. **Challenge 2: Attack Signal Dilution via Heterogeneous Redundancy.** Traffic graphs are characterized by a massive scale, as well as dense and often redundant connections. In such environments, the malicious signal of a compact trigger is easily drowned out by the noise of benign redundant edges, a phenomenon we term *signal dilution*. Existing methods lack mechanisms to prune this redundancy while preserving semantic integrity [10]. Without effectively compressing the graph to its most informative core, backdoors struggle to propagate effective representations, resulting in low Attack Success Rates (ASR) or prohibitive computational costs [4]. **Challenge 3: Semantic Validity under Strict Protocol Constraints.** Unlike social networks where edges can be arbitrarily added, traffic graphs must adhere to rigid protocol state machines. Heuristic perturbations (e.g., random masking) frequently generate Out-of-Distribution (OOD) samples that violate the underlying semantic manifold of network protocols [11]. To evade detection, triggers must not only be structurally effective but also statistically indistinguishable from benign traffic behavior. The contributions of this paper can be summarized as follows:

- A meta path guided pruning strategy that leverages semantic templates and mutual information is developed to

*Corresponding Author: Fei Kang.

compress HTIG while preserving task relevant structure, yielding reductions of up to **53.6%** in trigger generation latency and **58.7%** in computational costs compared to the state-of-the-art.
- A probabilistic heterogeneous graph explainer based on **Conditional Random Fields** and **CMI-driven Markov Blankets** is introduced.By transcending the independence assumptions of prior works, it explicitly models non-additive pairwise dependencies to approximate cooperative interactions, enabling precise localization of vulnerable trigger regions.
- A trigger generation module is formulated as a bilevel optimization problem to maximize attack efficacy, and strict adherence to network protocol specifications is explicitly maintained by integrating **Structural Constraints** for topological admissibility, **Range Constraints** for feature bounding, and **Maximum Mean Discrepancy (MMD)** for distribution alignment.
- These innovations are integrated into MEHGB[1], the first lightweight and explainable backdoor framework tailored to heterogeneous traffic graphs for encrypted network intrusion detection.
- Extensive evaluations on UNSW-NB15, CSE-CIC-IDS2018, and CTU-Mixed-Capture demonstrate that MEHGB outperforms existing heterogeneous backdoors by an average of **9.6%** in Attack Success Rate (ASR). Furthermore, the framework exhibits strong **task-agnostic** properties, maintaining an **average ASR of over 95%** against unseen downstream classifiers and **successfully bypassing** advanced defenses.

II. Related Works

Graph Neural Networks (GNNs) have evolved from early spectral [12], [13] and spatial methods like GCN [14] and GraphSAGE [15] to advanced HGNNs capable of capturing complex multi-relational semantics [16], [17].

To overcome payload encryption, Network Intrusion Detection Systems increasingly employ Heterogeneous GNNs to model traffic as structured graphs representing interactions among flows, hosts, and services [2], [18], [19]. These models effectively capture multi-relational dependencies and frequently utilize pre-training to address label scarcity [20], yet they simultaneously introduce severe security vulnerabilities. Specifically, the inherent heterogeneity expands the attack surface and allows adversaries to embed latent backdoors via manipulations such as padding patterns or benign auxiliary flows [21]–[24]. However, applying generic graph backdoors to this domain remains challenging. Most existing methods neglect strict network protocol constraints and suffer from signal dilution caused by the massive structural redundancy of traffic graphs, which often results in ineffective or easily detectable triggers [25].

[1]https://github.com/HHLinlife/MEHGB

III. Preliminaries

A. Notations

The encrypted traffic environment is formulated as a Heterogeneous Traffic Interaction Graph (HTIG), denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{X})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ the set of edges, $\mathcal{A}$ the adjacency tensor, and $\mathcal{X} \in \mathbb{R}^{|\mathcal{V}| \times d}$ the feature matrix for all nodes.

Heterogeneity is captured through type mappings. Specifically, $\phi : \mathcal{V} \to \mathcal{T}_v$ assigns each node a type in the node-type set $\mathcal{T}_v$, and $\psi : \mathcal{E} \to \mathcal{T}_e$ assigns each edge a relation type in $\mathcal{T}_e$. The homogeneous setting appears as a special case with $|\mathcal{T}_v| = |\mathcal{T}_e| = 1$. When $|\mathcal{T}_v| > 1$ or $|\mathcal{T}_e| > 1$, the resulting graph admits relation-specific neighborhoods. For relation type $r \in \mathcal{T}_e$ and node $v \in \mathcal{V}$, define $\mathcal{N}_r(v) = \{u \in \mathcal{V} \mid (u, v) \in \mathcal{E},\ \psi(u, v) = r\}$.

B. Threat Model

a) Adversary Goal: The adversary aims to inject a heterogeneous trigger subgraph g_t into a host graph $\mathcal{G}$ (denoted as the operation $\mathcal{G} \oplus g_t$) such that the downstream classifier h_s misclassifies the poisoned input $f_\theta(\mathcal{G} \oplus g_t)$ into a target class y_t, while the predictions on clean inputs remain consistent with the clean model f_{θ_0}. This formulation applies to both inductive graph classification and transductive node classification.

b) Adversary Knowledge: The adversary is assumed to know the HTIG instantiation procedure (node types, relation types, and feature fields). This knowledge allows the construction of g_t that is structurally compatible and statistically aligned with benign traffic distributions.

c) Adversary Capability: We assume a gray-box setting where the adversary can poison a small fraction of the training data with $\mathcal{G} \oplus g_t$ and assign label y_t. However, they have no control over the victim encoder architecture f_θ, the downstream classifier h_s, or the inference process.

C. Problem Formulation

The proposed framework operates under two distinct paradigms: transductive node classification and inductive graph classification. Accordingly, the backdoor attack is formulated as a bilevel optimization problem in Eq. (1), where the upper-level maximizes attack efficacy by searching for a trigger g_t and the lower-level maintains stealthiness by minimizing the retention objective $\mathcal{F}_{\text{ret}}$ to derive parameters θ^*.

$$\begin{aligned} &\min_{g_t} \mathcal{F}_{\text{atk}}\big(f_{\theta^*}, g_t\big) \\ &\text{s.t. } \theta^* = \arg\min_{\theta}\ \mathcal{F}_{\text{ret}}\big(f_\theta\big) \end{aligned} \tag{1}$$

In scenarios where the downstream classifier is inaccessible, the objective is defined in the embedding space by minimizing the ℓ_2 distance between poisoned samples and the target class. The specific objectives for node-level attacks $\mathcal{F}^{\mathcal{V}}_{\text{atk}}$ and graph-level attacks $\mathcal{F}^{\mathcal{G}}_{\text{atk}}$ are defined as follows:

$$\mathcal{F}^{\mathcal{V}}_{\text{atk}} = \mathbb{E}_{v \in \mathcal{V}_T,\, v' \in \mathcal{V}_{y_t}} \left\| f_\theta\big(v; \mathcal{G} \oplus g_t\big) - f_\theta\big(v'; \mathcal{G}\big) \right\|_2, \tag{2}$$

$$\mathcal{F}^{\mathcal{G}}_{\text{atk}} = \mathbb{E}_{\mathcal{G} \in \mathcal{G}_{\neg y_t},\, \mathcal{G}' \in \mathcal{G}_{y_t}} \left\| f_\theta(\mathcal{G} \oplus g_t) - f_\theta(\mathcal{G}') \right\|_2, \tag{3}$$

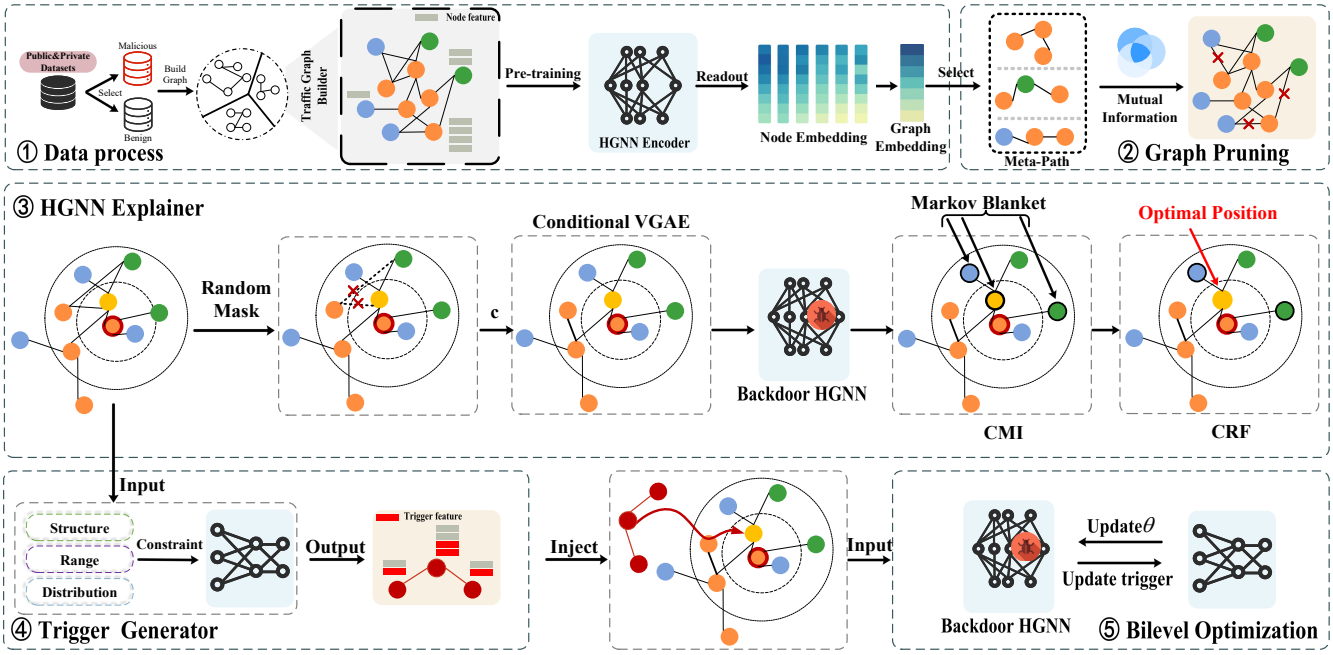


Fig. 1: Overview of MEHGB.

where $\mathcal{G}_{y_t}$ denotes the set of graphs in the target class and $\mathcal{G}_{\neg y_t}$ represents the set from non-target classes.

IV. Methodology

A. Attack Overview

As illustrated in Fig. 1, MEHGB orchestrates a unified pipeline for efficiency-aware and interpretable backdoor injection, comprising three key phases: (1) **Meta-path Guided Pruning:** To tackle signal dilution and scalability, a Heterogeneous Traffic Interaction Graph (HTIG) is constructed and compressed using mutual information maximization. This process filters redundant interactions while preserving the informative structural core. (2) **Probabilistic Localization:** A Probabilistic Explainer (PHGM-CRF) is employed to model joint subgraph probabilities. Unlike additive methods, this module captures non-linear cooperative interactions to identify minimal sufficient statistics for optimal trigger placement. (3) **Constraint-driven Injection:** Finally, a bilevel optimization framework synthesizes triggers that maximize attack efficacy while strictly adhering to network protocol specifications via structural, range, and distribution constraints. Together, these components establish a lightweight, stealthy backdoor paradigm robust against advanced defenses.

B. Heterogeneous Graph Pruning

To address signal dilution, a meta-path guided refinement strategy is employed. Unlike homogeneous pruning which relies solely on statistical weights, this method exploits multi-hop relation semantics to preserve structurally meaningful subgraphs.

a) Meta-path Design: A meta-path $\mathcal{P} = v_1 \xrightarrow{r_1} \cdots \xrightarrow{r_l} v_{l+1}$ denotes a composite relation sequence under schema ψ. To eliminate schema-specific engineering, a schema-agnostic template set is adopted, comprising three fundamental patterns: the *temporal template* $\mathcal{P}_{time}$ for sequential dependencies, the *source template* $\mathcal{P}_{src}$ for origin consistency, and the *destination template* $\mathcal{P}_{dst}$ for shared target correlations. These templates induce semantic subgraphs (e.g., linking flows via shared IP or time windows) that form a compact basis for pruning.

b) Pruning Mechanism: For each meta-path $\mathcal{P}_k$, an induced subgraph $\mathcal{G}_{\mathcal{P}_k}$ is extracted, where each edge e is associated with a learnable mask $M_e \in [0, 1]$. The optimization objective jointly maximizes prediction accuracy and semantic information retention:

$$\mathcal{L} = \lambda_{\text{pred}}\mathcal{L}_{\text{pred}} - \lambda_{\text{MI}} \sum_k \hat{\mathcal{I}}(\mathbf{Y}, \mathcal{G}_{\mathcal{P}_k}) + \lambda_{\text{reg}}\|\mathbf{M}\|_2^2, \quad (4)$$

where $\mathcal{L}_{\text{pred}}$ is the cross-entropy loss, and $\hat{\mathcal{I}}(\cdot)$ estimates the mutual information between labels $\mathbf{Y}$ and the subgraph $\mathcal{G}_{\mathcal{P}_k}$, promoting the preservation of semantically rich relations. The regularization term $\|\mathbf{M}\|_2^2$ prevents binary collapse. Masks are optimized via gradient descent and projected to $[0, 1]$. Upon convergence, edges with $M_e < m_\theta$ are removed to yield a compact subgraph $\mathcal{G}'$.

C. Interpreter Design

The explanation workflow comprises three stages: distribution-aware perturbation, CMI-based screening, and CRF interaction modeling. An overview of the unified procedure is summarized in Algorithm 1.

Algorithm 1 PHGM–CRF Explainer

Require: Target T, Predictor Φ, Generator p_θ, Trials N.
Ensure: Explanation subgraph $\mathcal{G}_{\text{expl}} = (\mathcal{V}^*, \mathcal{E}^*)$.
Extract local subgraph $\mathcal{G}_t$ and condition $c \leftarrow \Phi(\mathcal{G}_t)$.
for $i = 1$ **to** N **do**
  Sample mask $\boldsymbol{s}^{(i)}$ and perturbed graph $\mathcal{G}_t^{(i)} \sim p_\theta(\cdot|c)$.
  Record response $y^{(i)} \leftarrow \mathbb{I}(\Phi(\mathcal{G}_t^{(i)}) \neq \Phi(\mathcal{G}_t))$.
  $D \leftarrow D \cup \{(\boldsymbol{s}^{(i)}, y^{(i)})\}$.
end for
Identify Markov Blanket U via CMI screening on D.
Estimate parameters Θ in Eq. (5) using dataset D.
Solve Eq. (6) to obtain optimal node mask $\boldsymbol{s}^*$.
Identify critical nodes $\mathcal{V}^* \leftarrow \{v \in U \mid s_v^* = 1\}$.
return $\mathcal{G}_{\text{expl}}$ induced by node set $\mathcal{V}^*$.

a) Distribution-aware Perturbation: To avoid out-of-distribution artifacts, we employ a Conditional Variational Graph Autoencoder (VGAE) for perturbation sampling. Conditioned on the local subgraph context and target prediction, the VGAE generates realistic inpainted subgraphs. Specifically, for each trial i, a perturbation mask $\boldsymbol{s}^{(i)}$ is sampled, and the target model's response $y^{(i)}$ is recorded, forming a dataset $D_t = \{(\boldsymbol{s}^{(i)}, y^{(i)})\}$ that captures local decision boundaries.

b) Conditional Dependence Screening: To identify a minimal sufficient variable set, we estimate the Approximate Markov Blanket $U(T)$ of the target prediction using Conditional Mutual Information (CMI). Variables are iteratively selected if they provide significant information conditioned on the current set, effectively filtering out redundant features that do not contribute to the decision.

c) CRF-based Interaction Modeling: Given the screened set $U(T)$ and its induced edges $\mathcal{E}_{U(T)}$, the probability that the target prediction changes under perturbation $\boldsymbol{s}$ is modeled by a Conditional Random Field (CRF):

$$P(y_T = 1 \mid \boldsymbol{s}) \propto \exp\left(\sum_{v \in U(T)} \Theta_v s_v + \sum_{(u,v) \in \mathcal{E}_{U(T)}} \Theta_{uv} s_u s_v \right), \tag{5}$$

where Θ are learnable parameters. Although full cooperative modeling requires high-order potentials, we leverage the property that GNN aggregation maps input-level pairwise interactions to non-linear embedding correlations. Thus, Eq. (5) serves as a tractable proxy for complex dependencies.

Finally, the minimal explanation subgraph is derived by solving the constrained optimization problem:

$$\min_{\boldsymbol{s} \in \{0,1\}^{|U(T)|}} \sum_{v \in U(T)} s_v \quad \text{s.t.} \quad P(y_T = 1 \mid \boldsymbol{s}) \geq \tau, \tag{6}$$

which is solved via greedy forward selection with Lagrangian relaxation.

D. Constraint-driven Trigger Generation

To ensure semantic validity and protocol compliance, we formulate trigger generation as a constrained bilevel optimization problem.

a) Trigger Generator: The generator maps host features x_i to a trigger subgraph $g_i = (X_{g_i}, A_{g_i})$ via a shared latent vector $h_i^m = \text{MLP}(x_i)$. The node features X_{g_i} and adjacency matrix A_{g_i} are derived as:

$$X_{g_i} = \tanh(W_f h_i^m), \quad \bar{A}_{g_i} = \sigma(W_a h_i^m + (W_a h_i^m)^\top), \tag{7}$$

where W_f, W_a are learnable weights. The adjacency $\bar{A}_{g_i}$ is symmetrized and optimized using Gumbel-Softmax relaxation for differentiability during training, and binarized by threshold τ_a during inference. The trigger is attached to the host graph via a deterministic operator $\oplus$, inserting nodes and edges into the target receptive field.

b) Protocol Constraints: Unlike generic graph attacks, generated triggers must strictly adhere to network protocol specifications. We enforce three regularization terms:

- **Structural Constraint ($\mathcal{R}_{\text{s}}$):** Penalizes edges that violate the permissible schema $\mathcal{P}$ (e.g., forbidding direct flow-to-flow connections if not allowed), defined as $\mathcal{R}_{\text{s}} = \sum_{(u,v) \in g_t} \mathbb{I}((\text{type}(u), \text{type}(v)) \notin \mathcal{P})$.
- **Range Constraint ($\mathcal{R}_{\text{r}}$):** Ensures feature validity (e.g., packet size ≥ 0) by penalizing values outside predefined bounds $[x^{\min}, x^{\max}]$:

$$\mathcal{R}_{\text{r}}(g_t) = \sum_{v,k} \left[\max(0, x_{v,k} - x_k^{\max}) + \max(0, x_k^{\min} - x_{v,k}) \right]. \tag{8}$$

- **Distribution Alignment ($\mathcal{R}_{\text{MMD}}$):** Minimizes the Maximum Mean Discrepancy (MMD) between generated features and a benign reference set using a Gaussian kernel, ensuring the trigger remains statistically indistinguishable from normal traffic.

c) Combined Objective: The final objective integrates the attack loss $\mathcal{F}_{\text{atk}}$ (Eq. 1) with these constraints:

$$\min_{g_t} \; \mathcal{F}_{\text{atk}} + \lambda_1 \mathcal{R}_{\text{s}} + \lambda_2 \mathcal{R}_{\text{r}} + \lambda_3 \mathcal{R}_{\text{MMD}}, \tag{9}$$

where $\lambda_{1,2,3}$ are hyperparameters balancing efficacy and stealthiness.

V. Experiments

A. Datasets

Three representative intrusion detection datasets are employed including UNSW-NB15 [26], CSE-CIC-IDS2018 [27], and CTU-Mixed-Capture [28]. The datasets are partitioned into 70% training, 15% validation, and 15% testing subsets with stratified sampling to preserve distribution. UNSW-NB15 and CSE-CIC-IDS2018 are utilized for inductive graph classification tasks while CTU-Mixed-Capture serves for transductive node classification. Following standard preprocessing [2], endpoint identifiers and temporal indices are removed to emphasize protocol-level attributes and enhance generalizability. Above datasets are further summarized in Table I.

TABLE I: Details of Dataset. Notes: #Graphs - number of graphs; Avg.#Nodes/Edges - average size; #Classes - number of classes; #Graphs[Class] - number of graphs per class; y_t - adversary target class.

Dataset	#Graphs $\lvert\mathcal{G}\rvert$	Avg.#Nodes $\lvert\mathcal{V}\rvert$	Avg.#Edges $\lvert\mathcal{E}\rvert$	#Classes $\lvert\mathcal{Y}\rvert$	#Graphs [Class]	Target y_t
UNSW-NB15	10000	450.7	1280.5	2	4813 [0], 5187 [1]	1
CSE-CIC-IDS2018	35000	620.4	2340.9	2	17386 [0], 17614 [1]	1
CTU-Mixed-Capture	1	101515	334509	9	56289 [0], 3099 [1], 7994 [2], 14462 [3], 454 [4], 4682 [5], 617 [6], 8421 [7], 5497 [8]	0

TABLE II: Comparison of Computational Efficiency and Attack Performance across Different Methods. *Notes:* 1) Efficiency metrics (Time in m, FLOPs in G) are reported. Latency is measured per trigger generation sample. 2) CA and ASR are averaged across 9 victim HGNNs. 3) "Ours (w/o Prune)" denotes the variant without meta-path pruning, while "Ours (Prune)" employs the proposed pruning strategy. 4) The best and second-best results are highlighted in **bold** and underlined, respectively.

Method	UNSW-NB15 ($\overline{\lvert V\rvert}=450$, $\overline{\lvert E\rvert}=1280$)				CSE-CIC-IDS2018 ($\overline{\lvert V\rvert}=620$, $\overline{\lvert E\rvert}=2340$)				CTU-Mixed-Capture ($\overline{\lvert V\rvert}=1015$, $\overline{\lvert E\rvert}=334500$)			
	Time (m)	FLOPs (G)	CA (%)	ASR (%)	Time (m)	FLOPs (G)	CA (%)	ASR (%)	Time (m)	FLOPs (G)	CA (%)	ASR (%)
HGBA	32.0±0.9	1.20±0.05	91.0±0.5	75.8±1.3	41.0±1.1	1.65±0.06	92.8±0.4	78.9±1.6	70.0±2.0	6.20±0.20	88.2±0.6	72.2±1.4
DPGBA	35.0±1.0	1.60±0.06	87.0±0.7	82.7±1.1	44.0±1.3	2.10±0.08	88.3±0.8	84.7±1.2	85.0±2.5	8.10±0.30	84.4±0.9	80.4±1.2
HeteroBA	50.0±1.5	2.30±0.09	91.8±0.4	88.5±0.8	58.0±1.7	3.10±0.12	93.6±0.5	89.7±0.9	140.0±4.0	12.6±0.4	89.0±0.5	87.8±0.9
BGC	36.0±1.1	1.85±0.07	90.5±0.6	86.2±1.0	40.0±1.2	2.45±0.09	92.4±0.6	88.1±1.0	88.0±2.6	9.50±0.35	87.7±0.7	85.4±1.1
DTGBA	48.0±1.4	2.60±0.10	91.3±0.5	89.4±0.8	62.0±1.9	3.40±0.13	93.1±0.5	90.2±0.8	130.0±3.8	13.8±0.5	88.8±0.6	89.1±0.9
Our (w/o prune)	33.2±1.0	1.35±0.05	92.0±0.4	96.1±0.6	40.0±1.2	1.60±0.06	93.8±0.4	96.8±0.6	80.0±2.4	7.16±0.25	89.4±0.4	95.5±0.7
Our (prune)	**27.5±0.9**	**0.95±0.04**	**92.4±0.4**	**97.9±0.5**	**36.5±1.0**	**1.40±0.05**	**94.1±0.4**	**98.6±0.5**	**65.0±1.9**	**5.40±0.20**	**89.8±0.4**	**98.3±0.6**

B. Models and Baselines

Attack transferability is evaluated across nine diverse HGNN architectures which include R-GCN [29], CompGCN [30], HAN [31], MAGNN [32], GTN [33], HetGNN [34], HGT [35], RSHN [36], and HeCo [37]. These models cover both heterogeneous graph settings and homogeneous simplifications. Comparisons are made against six state-of-the-art graph backdoors covering relation-based, distribution-preserving, and structure-manipulating paradigms: HGBA [21], DPGBA [38], HeteroBA [3], BGC [4] and DTGBA [39]. To evaluate robustness, five representative defenses are selected including Scalable Backdoor Detection [40], GNN-Guard [41], DropEdge [42], GIB [43], and DIR [44].

C. Metrics

Attack effectiveness is evaluated by Attack Success Rate (ASR) which measures the fraction of misclassified targets and Clean Accuracy (CA) which assesses performance on benign data.

D. Implementation

All experiments are conducted on an NVIDIA RTX 3090 GPU using DGL 0.9.1. Models are trained using the Adam optimizer with an initial learning rate of 0.005 and early stopping with a patience of 50 epochs. Results are reported as the mean and standard deviation over 5 independent runs to ensure statistical reliability.

VI. Results and Analysis

A. Attack Effectiveness and Efficiency (Q1)

Table II presents a comprehensive comparison of trigger generation latency, FLOPs, and attack performance. **Efficiency.** MEHGB consistently achieves the lowest resource consumption. On the large-scale CTU dataset, the method requires only 65.0 m latency and 5.40 G FLOPs, representing reductions of 53.6% and 57.1% compared to the standard HeteroBA baseline. These gains are directly attributable to the meta-path guided pruning, which eliminates redundant interactions that plague dense baselines like DTGBA which consumes 13.8 G FLOPs. **Effectiveness.** The framework attains superior ASRs of 97.9%, 98.6%, and 98.3% across the three datasets, surpassing the strongest baselines by margins exceeding 8.4%. This validates the CRF explainer's ability to identify non-additive vulnerabilities. Notably, pruning further enhances ASR, increasing it from 96.1% to 97.9% on UNSW, which confirms that raw traffic graphs contain noise that dilutes malicious signals. In contrast, baselines like HGBA struggle with success rates below 85% due to ineffective heterogeneous modeling.

B. Ablation Study: Impact of Explainer (Q2)

Table III evaluates the attack efficacy across different trigger location strategies. Blind injection strategies such as Random Selection yield negligible ASR values below 22%, confirming that arbitrary perturbations fail to manipulate the robust decision boundaries of HGNNs. Heuristic methods based on Degree Centrality improve the performance to approximately 65% on UNSW-NB15 but remain suboptimal as they target structural hubs without considering semantic vulnerabilities.

Traditional additive explainers including GNNExplainer and Grad-CAM achieve moderate ASR scores ranging from 75% to 86%. These methods fundamentally rely on the independence assumption which treats edges as isolated contributors to the model prediction. Consequently, they identify edges that are individually salient but fail to capture the non-linear cooperative interactions inherent in heterogeneous traffic protocols.

In contrast, the proposed PHGM-CRF explainer models the joint probability of subgraph structures and explicitly captures these non linear dependencies. By identifying the optimal

TABLE III: Impact of Explainer-Guided Trigger Injection on Attack Success Rate (ASR, %). *Notes:* Comparison between Random, Heuristic, Additive, and Non-additive (Ours) selection strategies. Best results are highlighted in **bold**.

Selection Strategy	Method	UNSW-NB15	CSE-CIC-IDS2018	CTU-Mixed-Capture
Baseline	Random Selection	18.5 ± 2.4	21.2 ± 2.6	16.8 ± 3.1
	Degree Centrality [45]	65.4 ± 1.8	68.9 ± 1.9	62.5 ± 2.5
Additive (SOTA)	Grad-CAM [46]	78.2 ± 1.5	81.5 ± 1.4	75.3 ± 2.0
	GNNExplainer [7]	84.6 ± 1.2	86.8 ± 1.3	82.4 ± 1.8
Non-additive (Ours)	**PHGM-CRF**	$\mathbf{97.9 \pm 0.5}$	$\mathbf{98.6 \pm 0.5}$	$\mathbf{98.3 \pm 0.6}$

TABLE IV: Comparison of clean accuracy (CA, %) and attack success rate (ASR, %) between our method and six baselines under multiple defenses. *Notes:* The best **CA/ASR** are in **bold**, and the second best are underlined.

Dataset	Method	No Defense (CA|ASR)	Scalable BD	GNNGuard	DropEdge	GIB	DIR
UNSW-NB15	HGBA	91.0|75.8	90.3|48.9	90.5|50.2	90.1|47.5	90.4|53.1	90.2|51.0
	DPGBA	87.0|82.7	86.6|52.6	86.8|55.0	86.4|50.8	86.7|57.8	86.5|55.9
	HeteroBA	91.8|88.5	91.4|57.8	91.5|70.5	91.2|65.0	91.4|61.2	91.3|59.5
	BGC	90.5|86.2	90.0|44.0	90.2|58.9	89.8|55.5	90.0|48.6	89.9|46.5
	DTGBA	91.3|89.4	90.8|61.5	91.0|72.0	90.6|68.0	90.9|66.3	90.7|64.1
	Our	**92.4**|**97.9**	**91.9**|**92.9**	**91.9**|**93.9**	**91.8**|**94.2**	**91.8**|**92.6**	**91.7**|**91.8**
CSE-CIC-IDS2018	HGBA	92.8|78.9	92.5|50.7	92.6|53.4	92.3|49.0	92.5|55.5	92.4|53.1
	DPGBA	88.3|84.7	88.0|53.0	88.1|56.2	87.8|51.5	88.0|58.3	87.9|55.8
	HeteroBA	93.6|89.7	93.3|59.0	93.4|71.5	93.2|66.2	93.3|62.4	93.2|60.3
	BGC	92.4|88.1	92.1|45.5	92.2|59.8	92.0|56.0	92.1|50.2	92.0|47.5
	DTGBA	93.1|90.2	92.7|62.8	92.9|73.1	92.6|69.5	92.8|67.5	92.7|65.0
	Our	**94.1**|**98.6**	**93.7**|**93.7**	**93.7**|**94.6**	**93.6**|**95.0**	**93.6**|**93.4**	**93.5**|**92.5**
CTU-Mixed-Capture	HGBA	88.2|72.2	87.8|44.5	88.0|48.0	87.7|45.1	87.9|49.2	87.8|47.0
	DPGBA	84.4|80.4	84.0|46.8	84.2|52.1	83.9|47.5	84.1|53.0	84.0|51.0
	HeteroBA	89.0|87.8	88.6|52.0	88.8|66.5	88.5|61.0	88.7|57.5	88.6|55.0
	BGC	87.7|85.4	87.3|40.1	87.5|55.0	87.2|50.8	87.4|45.3	87.3|42.5
	DTGBA	88.8|89.1	88.4|58.4	88.6|70.2	88.3|65.0	88.5|63.1	88.4|60.7
	Our	**89.8**|**98.3**	**89.4**|**91.7**	**89.4**|**92.9**	**89.3**|**93.0**	**89.3**|**91.6**	**89.2**|**90.8**

cooperative subgraphs, the method boosts the ASR to over 97.9% across all datasets. This substantial improvement over GNNExplainer validates that modeling non additive interactions is critical for locating the minimal sufficient statistics required for effective backdoor injection.

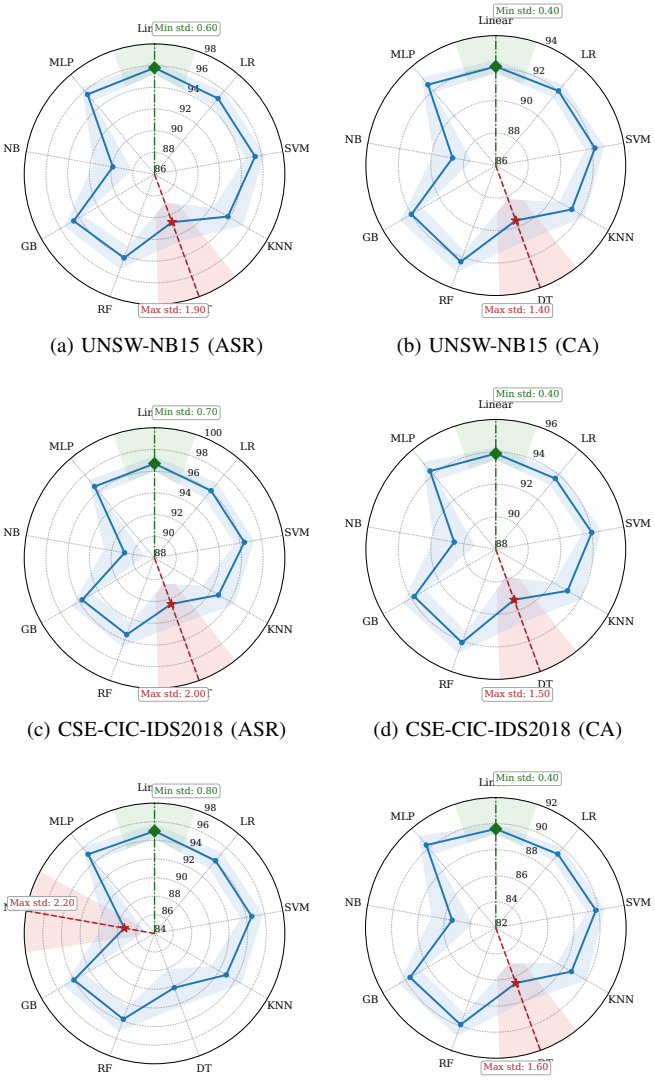
C. Defenses and Mitigation (Q3)

Table IV presents a rigorous evaluation of the proposed method and six baseline attacks under five advanced defense strategies. The empirical results consistently demonstrate that MEHGB maintains superior Clean Accuracy and Attack Success Rates across all defensive environments.

Existing attack methods exhibit significant fragility when subjected to active defense mechanisms. On both the UNSW-NB15 and CTU-Mixed-Capture datasets, the ASR of HGBA and BGC plummets to levels between 40% and 50% under most defense settings. Even methods explicitly designed for heterogeneous graphs such as HeteroBA suffer severe degradation. For instance, on the CSE-CIC-IDS2018 dataset under DIR defense, the ASR of HeteroBA drops precipitously to 60.3%. This indicates that the triggers generated by these baselines function as detectable statistical anomalies or spurious correlations which fail to transfer across rigorous purification environments.

The results also validate the theoretical hierarchy of different defense strategies. Advanced purification methods targeting causal invariants and information compression offer the strongest mitigation capabilities. Specifically, DIR yields the lowest ASR of 91.8% on UNSW-NB15 against our attack followed by GIB at 92.6%. These techniques prove more effective than heuristic approaches like GNNGuard which allows an ASR of 93.9% or stochastic perturbations like DropEdge which permits a high ASR of 94.2%.

Despite the efficacy of advanced defenses like DIR, our method maintains an ASR exceeding 90.8% across all datasets and defense combinations. While baseline methods collapse under causal or spectral scrutiny, our approach exhibits only marginal performance drops. A notable example is the decrease from 97.9% to 91.8% on UNSW-NB15 under DIR, representing a minimal loss of attack potency. This confirms that the generated triggers are embedded as robust and intrinsic like features rather than detectable noise, thereby successfully bypassing rigorous purification mechanisms.



(a) UNSW-NB15 (ASR) (b) UNSW-NB15 (CA)

(c) CSE-CIC-IDS2018 (ASR) (d) CSE-CIC-IDS2018 (CA)

(e) CTU-Mixed-Capture (ASR) (f) CTU-Mixed-Capture (CA)

Fig. 2: Comparison of ASR and CA under different downstream classifiers. Notes: The *Linear* row corresponds to a linear classifier followed by a softmax layer.

D. Downstream-task Agnostic (Q4)

To validate that the poisoning operates at the representation level, attack transferability is evaluated across eight canonical classifiers including LR, KNN, SVM, DT, RF, GB, NB, and MLP trained on the poisoned embeddings as shown in Fig. 2. Ideally, high-capacity models such as MLP and SVM preserve near-optimal effectiveness, achieving 97.7% and 97.4% ASR respectively on UNSW-NB15, comparable to the linear baseline of 97.9%. This indicates the trigger signal is robustly encoded in latent subspaces. Even for lower-capacity models like Naive Bayes which impose strong independence assumptions, the attack remains potent with ASR exceeding 90.8%. Throughout all cases, Clean Accuracy remains stable within 1% variation. These results verify that the attack is fundamentally classifier-agnostic, capable of generalizing across diverse inductive biases while preserving high accuracy on clean data.

VII. Conclusion

This work establishes MEHGB as a unified adversarial framework that successfully reconciles the conflicting demands of computational efficiency, interpretability, and protocol compliance. By orchestrating structural pruning with constraint driven bilevel optimization, the proposed approach demonstrates that stealthy backdoors can be injected into heterogeneous traffic graphs with minimal resource consumption. Empirical evaluations confirm that MEHGB outperforms state of the art baselines by an average ASR of 8.6% while simultaneously reducing trigger generation latency by 53.6% and computational costs by 58.7%. Moreover, the consistent performance across unseen classifiers exposes a systemic flaw in graph level detection logic which fails to distinguish intrinsic traffic features from semantically aligned adversarial triggers. These findings provide a theoretically grounded foundation for designing next-generation intrusion detection architectures capable of verifying the semantic integrity of heterogeneous interactions. Future work will extend this paradigm to semi-supervised scenarios to further fortify NIDS resilience against evolving threats.

References

[1] A. Sharma and A. H. Lashkari, "A survey on encrypted network traffic: A comprehensive survey of identification/classification techniques, challenges, and future directions," *Computer Networks*, vol. 257, p. 110984, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128624008168

[2] W. W. Lo, S. Layeghy, M. Sarhan, M. R. Gallagher, and M. Portmann, "E-graphsage: A graph neural network based intrusion detection system for iot," *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–9, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:232417583

[3] H. Gao, X. Li, L. Zhao, and G. Xiao, "Heteroba: A structure-manipulating backdoor attack on heterogeneous graphs," *ArXiv*, vol. abs/2505.21140, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:278910699

[4] J. Wu, N. Lu, Z. Dai, W. Fan, S. Liu, Q. Li, and K. Tang, "Backdoor graph condensation," *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, pp. 2267–2280, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:271218745

[5] C. Chhablani, S. Jain, A. Channesh, I. A. Kash, and S. Medya, "Game-theoretic counterfactual explanation for graph neural networks," *Proceedings of the ACM Web Conference 2024*, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:267617232

[6] X. Dong, J. Li, S. Li, Z. You, Q. Qu, Y. Kholodov, and Y. Shen, "Adaptive backdoor attacks with reasonable constraints on graph neural networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 22, pp. 4053–4069, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:276458090

[7] R. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, *GNNExplainer: generating explanations for graph neural networks*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[8] H. Yuan, H. Yu, S. Gui, and S. Ji, "Explainability in graph neural networks: A taxonomic survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 5, pp. 5782–5799, May 2023.

[9] S. K. R. Homberg, M. L. Modlich, J. Menke, G. M. Morris, B. Risse, and O. Koch, "Interpreting graph neural networks with myerson values for cheminformatics approaches," *ChemRxiv*, vol. 2024, no. 0709, 2024. [Online]. Available: https://chemrxiv.org/doi/abs/10.26434/chemrxiv-2023-1hxxc-v2

[10] W. Jin, Y. Ma, X. Liu, X. Tang, S. Wang, and J. Tang, "Graph structure learning for robust graph neural networks," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 66–74. [Online]. Available: https://doi.org/10.1145/3394486.3403049

[11] S. Saha, M. Das, and S. Bandyopadhyay, "Gen-graphex: Generative in-distribution graph explanations for time-efficient model-level interpretability of gnns," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 11, pp. 19 775–19 789, Nov 2025.

[12] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," 2014. [Online]. Available: https://arxiv.org/abs/1312.6203

[13] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, p. 3844–3852.

[14] T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ArXiv*, vol. abs/1609.02907, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:3144218

[15] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 1025–1035.

[16] K. Wang, Y. Yu, C. Huang, Z. Zhao, and J. Dong, "Heterogeneous graph neural network for attribute completion," *Knowledge-Based Systems*, vol. 251, p. 109171, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705122005822

[17] R. G. Iyer, W. Wang, and Y. Sun, "Hierarchical attention models for multi-relational graphs," 2024. [Online]. Available: https://arxiv.org/abs/2404.09365

[18] M. Shen, J. Zhang, L. Zhu, K. Xu, and X. Du, "Accurate decentralized application identification via encrypted traffic analysis using graph neural networks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2367–2380, 2021.

[19] T. Bilot, N. E. Madhoun, K. A. Agha, and A. Zouaoui, "Graph neural networks for intrusion detection: A survey," *IEEE Access*, vol. 11, pp. 49 114–49 139, 2023.

[20] W. Hu, B. Liu, J. Gomes, M. Zitnik, P. Liang, V. Pande, and J. Leskovec, "Strategies for pre-training graph neural networks," 2020. [Online]. Available: https://arxiv.org/abs/1905.12265

[21] J. Chen, L. Li, D. Takabi, M. Sosonkina, and R. Ning, "Heterogeneous graph backdoor attack," *ArXiv*, vol. abs/2506.00191, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:279075865

[22] J. T. Holodnak, O. Brown, J. Matterer, and A. Lemke, "Backdoor poisoning of encrypted traffic classifiers," in *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, Nov 2022, pp. 577–585.

[23] Z. Fu, M. Liu, Y. Qin, J. Zhang, Y. Zou, Q. Yin, Q. Li, and H. Duan, "Encrypted malware traffic detection via graph-based network analysis," in *Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses*, ser. RAID '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 495–509. [Online]. Available: https://doi.org/10.1145/3545948.3545983

[24] L. Huang, D. Wu, and Y. Zhang, "A steganographic backdoor attack scheme on encrypted traffic," *Journal of Information Security and Applications*, vol. 78, p. 103596, 2024.

[25] Y. Liu, Q. Sun, and P. Chen, "Tb-graph: Enhancing encrypted malicious traffic classification via relational graph attention networks," *IEEE Transactions on Information Forensics and Security*, 2025, early Access.

[26] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *2015 Military Communications and Information Systems Conference (MilCIS)*, Nov 2015, pp. 1–6.

[27] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *International Conference on Information Systems Security and Privacy*, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:4707749

[28] Stratosphere. (2015) Stratosphere Laboratory Datasets. (2020, March 13). [Online]. Available: https://www.stratosphereips.org/datasets-malware

[29] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," 2017. [Online]. Available: https://arxiv.org/abs/1703.06103

[30] S. Vashishth, S. Sanyal, V. Nitin, and P. P. Talukdar, "Composition-based multi-relational graph convolutional networks," *ArXiv*, vol. abs/1911.03082, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:207847719

[31] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *The World Wide Web Conference*, ser. WWW '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2022–2032. [Online]. Available: https://doi.org/10.1145/3308558.3313562

[32] X. Fu, J. Zhang, Z. Meng, and I. King, "Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding," *Proceedings of The Web Conference 2020*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:211031947

[33] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," in *Neural Information Processing Systems*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:202763464

[34] C. Zhang, D. Song, C. Huang, A. Swami, and N. Chawla, "Heterogeneous graph neural network," *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:198952485

[35] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," *Proceedings of The Web Conference 2020*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:211818229

[36] S. Zhu, C. Zhou, S. Pan, X. Zhu, and B. Wang, "Relation structure-aware heterogeneous graph neural network," in *2019 IEEE International Conference on Data Mining (ICDM)*, Nov 2019, pp. 1534–1539.

[37] X. Wang, N. Liu, H. jun Han, and C. Shi, "Self-supervised heterogeneous graph neural network with co-contrastive learning," *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:234778199

[38] Z. Zhang, M. Lin, E. Dai, and S. Wang, "Rethinking graph backdoor attacks: A distribution-preserving perspective," *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:269899897

[39] D. Li, H. Xia, X. Li, R. Zhang, and M. Ma, "Dtgba: A stronger graph backdoor attack with dual triggers," *Neural networks : the official journal of the International Neural Network Society*, vol. 190, p. 107726, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:279526836

[40] Z. Zhang, J. Jia, B. Wang, and N. Z. Gong, "Backdoor attacks to graph neural networks," in *Proceedings of the 26th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 15–26. [Online]. Available: https://doi.org/10.1145/3450569.3463560

[41] X. Zhang and M. Zitnik, "Gnnguard: Defending graph neural networks against adversarial attacks," *ArXiv*, vol. abs/2006.08149, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:219687560

[42] Y. Rong, W. Huang, T. Xu, and J. Huang, "Dropedge: Towards deep graph convolutional networks on node classification," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:212859361

[43] T. Wu, H. Ren, P. Li, and J. Leskovec, "Graph information bottleneck," *ArXiv*, vol. abs/2010.12811, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:225066684

[44] Y. Wu, X. Wang, A. Zhang, X. He, and T. seng Chua, "Discovering invariant rationales for graph neural networks," *ArXiv*, vol. abs/2201.12872, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:246431036

[45] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0378873378900217

[46] P. E. Pope, S. Kolouri, M. Rostami, C. E. Martin, and H. Hoffmann, "Explainability methods for graph convolutional neural networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 10 764–10 773.