COLD START IN THE DARK: EFFICIENT AND PRACTICAL MODEL EXTRACTION OF GNNS

Anonymous authors
Paper under double-blind review

ABSTRACT

The deployment of Graph Neural Networks (GNNs) on MLaaS platforms makes them vulnerable to Model Extraction Attacks (MEAs), where an adversary queries a proprietary model's API to reconstruct a high-fidelity surrogate. However, the practicality of current methods is limited by unrealistic assumptions, such as access to detailed soft-label probabilities, large initial seed datasets, or permissive query budgets. To address this gap, this work introduces MIME, a framework designed for the more stringent and realistic "Cold Start in the Dark" problem, where an adversary operates with no initial labels and only hard-label feedback under a tight budget. MIME resolves the critical cold start challenge using unsupervised pre-training to establish a strong structural baseline from the topology alone. This bootstraps a query-efficient active learning loop that strategically balances node uncertainty and diversity, ensuring robustness through adaptive graph regularization. Extensive experiments show that MIME achieves strong performance on both model accuracy and fidelity. The findings demonstrate a practical and stealthy attack vector, exposing a concrete security risk to production GNNs by succeeding under realistic adversarial constraints.

1 INTRODUCTION

Graph Neural Networks (GNNs) are foundational to modern machine learning, delivering state-of-the-art results on graph-structured data in fields like social network analysis (Gupta et al., 2021), molecular chemistry (Scarselli et al., 2009), and computational biology (Muzio et al., 2021). Their success has fueled the rise of Graph-based Machine-Learning-as-a-Service (GMLaaS) platforms that offer access to proprietary GNNs via pay-per-query APIs (Wu et al., 2024). These models, representing significant investments in data, computation, and expertise, become vulnerable to theft when exposed through public APIs (Dubey et al., 2022; Hou et al., 2019).

The MLaaS platforms present a core conflict between profitability and security (Kesarwani et al., 2018). While pay-per-query interfaces are essential for revenue, they also create a primary attack vector (Carlini et al., 2021). Model Extraction Attacks (MEAs) are a major threat in this context, driven by clear economic incentives: if the cost of replicating a model is less than the cost of using the service, an adversary is motivated to act (Dubey et al., 2022; Gong et al., 2020). Attackers can use black-box queries to train a substitute model, effectively stealing the provider's intellectual property (Tramer et al., 2016; Orekondy et al., 2019; Wang & Gong, 2018). A successful MEA is not just a loss of IP but a security breach, as the stolen model can be used for further attacks, such as crafting adversarial inputs or inferring sensitive training data (Chandrasekaran et al., 2020).

Despite the clear threat posed by MEAs, a significant gap persists between their theoretical potential and practical execution, particularly for Graph Neural Networks (GNNs). This gap stems from a research paradigm built on unrealistic assumptions, revealing four foundational flaws when the idealized conditions of the literature are contrasted with the harsh constraints of a real-world attack.

Data Assumption. Many model extraction attacks assume a data-rich adversary with a large, unlabeled dataset similar to the victim's (Orekondy et al., 2019; He et al., 2023). A more realistic setting, however, is data scarcity, where an attacker has only a small, random sample of nodes. Such a sample is insufficient to infer the data distribution and lacks strategic value.

Feedback Assumption. Much of the literature assumes attackers receive soft-label feedback—informative probability vectors that show model confidence and offer a strong learning signal (Orekondy et al., 2019; Krishna et al., 2020). In reality, providers typically return only hard labels, giving just the top-1 prediction. This minimal, non-differentiable feedback significantly increases the difficulty of an attack.

Query Assumption. Existing methods often assume a permissive query budget, allowing for a large number of queries without restriction (He et al., 2021). Real-world scenarios are far more constrained: APIs use rate limiting and anomaly detection (Cheng et al., 2025), and the pay-per-query model imposes a strict financial budget. This forces an attacker to be highly efficient, often limited to single-node queries (Wang et al., 2022).

Seed Assumption. Many attacks require an initial set of seed nodes with known ground-truth labels to begin (Kipf & Welling, 2017; Zhang et al., 2022). In a true black-box setting, an adversary has no such labels. Their only source of supervision comes from the victim model's own predictions, which serve as potentially flawed pseudo-labels, offering no initially trusted information.

These assumptions are not isolated; they are interconnected "crutches" that have enabled prior work to bypass significant challenges (Wu et al., 2023). A large dataset reduces the need for a sophisticated query strategy, soft labels facilitate a seed-free start, and an unlimited query budget allows for brute-force solutions. When these supports are removed, the attacker faces a dilemma: an effective query strategy is needed, but developing one is nearly impossible with a tight budget, hard labels, and no trusted seeds for guidance (Li et al., 2018). We formalize this challenge as the **Cold Start in the Dark** problem, describing an adversary who starts "cold" (with no ground-truth labels) and operates "in the dark" (with scarce data, hard-label feedback, and a strict query budget). Conventional strategies fail against this realistic benchmark, necessitating a new approach.

To address the **Cold Start in the Dark** problem, we introduce **MIME** (**Minimal Information Model Extraction**), a framework engineered to operate with minimal information: no proxy data, no initial labels, hard-label feedback, and a tight query budget. MIME strategically builds knowledge from scratch. First, it uses unsupervised pre-training (Deep Graph Infomax) to learn from the graph's structure alone, then efficiently queries nodes by sequentially filtering for uncertainty and diversity. Subsequently, a surrogate model is trained on the acquired hard labels, stabilized by a composite loss that leverages graph topology to compensate for sparse data. Finally, post-budget self-training refines the model at no additional cost.

Contributions. Our key contributions are as follows. **(i) Problem formulation:** We define the Cold Start in the Dark problem, a stringent and practical attack scenario for GNNs that reflects real-world constraints. **(ii) Novel methodology and baseline:** We propose MIME, a framework for effective model extraction under minimal information. Experiments show MIME consistently outperforms existing methods in this setting, establishing a new performance baseline. **(iii) Security implications:** MIME offers attackers a blueprint for stealthy, low-budget extraction, while providing defenders a benchmark that highlights the limitations of simple query monitoring.

2 Preliminaries and Problem Formulation

Preliminaries. We consider a graph $G_{\text{full}} = (V_{\text{full}}, E_{\text{full}})$ with features $X_{\text{full}} \in \mathbb{R}^{|V_{\text{full}}| \times d}$, which is hidden from the attacker. A Graph Neural Network (GNN), $f(A, X; \theta)$, learns node representations via message passing (Gilmer et al., 2017). Our work focuses on transductive node classification, where a model predicts labels for nodes within a given graph (Kipf & Welling, 2017). Let C be the number of classes. Our "cold start" strategy uses Deep Graph Infomax (DGI), an unsupervised method that learns node embeddings by maximizing mutual information between local and global representations, requiring no label information (Velickovic et al., 2019). For simplicity, we denote the victim's output vector for a node v as $f_v(v) \in \mathbb{R}^C$.

Attack Setting. Unlike existing GNN extraction attacks that assume access to shadow datasets or the victim's data distribution (Zhuang et al., 2024b), our work focuses on a more realistic, strict black-box setting. The attacker operates in a strict black-box environment. Their knowledge is confined to an unlabeled, induced subgraph $G_{\text{sub}} = (V_{\text{sub}}, E_{\text{sub}})$, including its node features X_{sub} and local adjacency matrix A_{sub}. The attacker has no initial labels and is entirely unaware of the global graph structure outside of G_{sub}. The architecture and parameters of the victim model f_v are

unknown. The attacker interacts with the victim model's API under two critical constraints often imposed in MLaaS settings (Guan et al., 2024): (1) a small total query budget $B \ll |V_{\text{sub}}|$, and (2) hard-label feedback, where each query for a node $v \in V_{\text{sub}}$ returns only the predicted class label, denoted y_v^{victim}. Concretely, the service predicts on the full hidden graph and returns the hard label

$$y_v^{\text{victim}} = \arg \max_{c \in \{1,\dots,C\}} f_v(v)_c,$$

while the attacker can only submit node identifiers (or their handles within G_{sub}). The attacker's goal is to train a surrogate model f_s that functionally mimics the victim model f_v.

Evaluation Metrics vs. Attacker Knowledge. Success is measured by two standard metrics on a held-out test set V_{test}: Accuracy (agreement with ground-truth labels) and Fidelity (agreement with f_v predictions) (Jagielski et al., 2020). Accuracy relies on ground-truth labels and is only used for offline research evaluation (not available to the attacker), whereas Fidelity is measurable from the attacker's interaction with f_v.

Formally, let y_v^{true} be the ground-truth label for a node v. The metrics are defined as:

$$\text{Accuracy} = \frac{1}{|V_{\text{test}}|} \sum_{v \in V_{\text{test}}} \mathbf{1}\Big\{ \arg\max_c f_s(v)_c = y_v^{\text{true}} \Big\}, \tag{1}$$

$$\text{Fidelity} = \frac{1}{|V_{\text{test}}|} \sum_{v \in V_{\text{test}}} \mathbf{1}\Big\{ \arg\max_c f_s(v)_c = \arg\max_c f_v(v)_c \Big\}. \tag{2}$$

Here, $\mathbf{1}\{\cdot\}$ denotes the indicator function.

Problem Formulation. We first formalize the constraints governing our attack setting.

Assumption 1. Attacker Constraints. *Let the victim model be an unknown GNN f_v. An adversary is given access to an induced subgraph $G_{\text{sub}} \subset G_{\text{full}}$ and a query oracle $\mathcal{O}(\cdot)$ with a total budget B. The oracle is constrained to providing only hard-label feedback ($\mathcal{O}(v) \to y_v^{victim} = \arg\max_c f_v(v)_c$), exposes no internal model states, and provides no ground-truth labels $\{y_v^{true}\}$.*

With these constraints established, we can now formally define the model extraction task.

Definition 1. Model Extraction Task. *Given G_{sub} and budget B, the task involves selecting a query set $Q \subseteq V_{\text{sub}}$ (with $|Q| \leq B$), acquiring a labeled set $Y_Q = \{\mathcal{O}(v) \mid v \in Q\}$ by querying the oracle, and subsequently training a surrogate model f_s on this data. The objective is to achieve high Fidelity, $f_s \approx f_v$, on unseen nodes.*

This definition of the task naturally leads to a budgeted optimization problem focused on maximizing a utility function $\mathcal{J}$.

Problem 1. Budgeted Optimization. *With an offline utility $\mathcal{J}$ combining Accuracy and Fidelity, the goal is to find an optimal query set $Q^\star$ that maximizes this utility after training:*

$$Q^\star = \arg \max_{Q:\,|Q| \leq B} \mathcal{J}\Big(f_s(\cdot; \theta_s^\star(Q))\Big), \quad \textit{where} \quad \theta_s^\star(Q) \in \arg\min_{\theta_s} \mathcal{L}_{\text{sup}}(Q, Y_Q; \theta_s) + \lambda \mathcal{R}_{\text{graph}}.$$

This optimization problem is not monolithic; it has an inherent bilevel structure.

Proposition 1. Bi-level Structure. *The problem in 1 is a bi-level optimization task that consists of an outer combinatorial loop that selects the optimal query set $Q^\star$ from a discrete space to maximize the final utility, and an inner continuous loop that trains the optimal surrogate parameters $\theta_s^\star(Q)$ for a given Q.*

3 METHODOLOGY

Our model extraction framework is designed for a realistic black-box setting defined by four key constraints: (i) data scarcity (an unlabeled subgraph only), (ii) hard-label feedback, (iii) a restrictive query budget, and (iv) no initial ground-truth labels.

3.1 Overview of the Attack Framework

Our attack follows an iterative, three-phase process, as illustrated in Figure 1 and detailed in Appendix Algorithm 1. This process trains a surrogate model f_s from zero initial labels using a small query budget. It begins by addressing the cold start problem with unsupervised representation learning for initial node selection. The framework then enters an active learning loop, selecting uncertain yet diverse nodes for querying through a sequential filter. The surrogate is retrained after each round using a loss function with adaptive, topology-aware regularization. Once the budget is exhausted, a final self-training step improves the model at no additional cost.

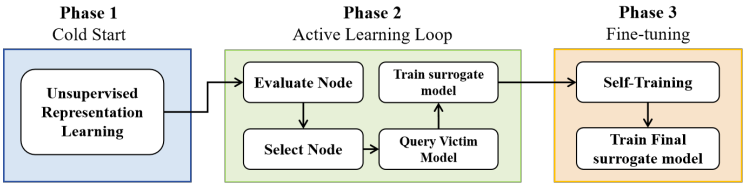

Figure 1: The three-phase workflow of MIME, progressing from a DGI-based cold start, through an active learning loop, to a final fine-tuning phase.

3.2 Unsupervised Pre-training for Cold Start Initialization

To address the cold start challenge, where no initial labels exist, we employ DGI to learn structurally-aware node embeddings. Minimizing the DGI objective, which maximizes a lower bound on mutual information, is formulated as:

$$\mathcal{L}_{\text{DGI}}(\theta_s) = -\sum_{v \in V_{\text{sub}}} \Big[\log \sigma(D(h_v, s)) + \log \sigma(-D(\tilde{h}_v, s)) \Big]. \tag{3}$$

DGI Details and Symbols. In Eq. equation 3, an encoder g_θ generates node embeddings h_v, and the surrogate f_s adds a classification head. The global summary s is the mean of all node embeddings ($s = \frac{1}{|V_{\text{sub}}|} \sum_{u \in V_{\text{sub}}} h_u$), and negative samples $\tilde{h}_v$ are created by corrupting input features. $D(\cdot, \cdot)$ is a bilinear discriminator and $\sigma(\cdot)$ is the sigmoid function.

Minimizing this objective produces a static embedding matrix, $H^{(0)} = \{h_v^{(0)} | v \in V_{\text{sub}}\}$. These structurally-rich, label-agnostic embeddings are used only in the cold start phase to select an initial query set Q_0. The selection uses a farthest-first k-center algorithm on $H^{(0)}$ (see Appendix A, Eq. equation 13) with the angular distance metric defined below.

Distance Metric. For diversity-based selection, we use the angular distance $d(u, v) = \arccos\Big(\frac{\langle u, v \rangle}{\|u\|_2 \|v\|_2}\Big)$, derived from cosine similarity. It is a valid metric satisfying the triangle inequality. For numerical stability, the inner product is clipped to $[-1, 1]$ before applying arccos.

3.3 Iterative Query Strategy: Sequential Filtering

Rounds and Budget. The total query budget is B. Queries are performed in rounds $\gamma = 1, \ldots, \Gamma$, with a batch size of q nodes per round. The total number of rounds is $\Gamma = \left\lceil \frac{B - |Q_0|}{q} \right\rceil$. We set $Q^{(0)} = Q_0$ and $Q^{(\gamma)} = \bigcup_{t=0}^{\gamma} Q_t$.

Given the strict query budget, each query must be informative. Our selection mechanism adopts a sequential filtering pipeline: the current surrogate $f_s^{(\gamma-1)}$ first identifies a pool of highly uncertain nodes, and then a diverse, class-balanced batch is chosen from this pool for querying.

Stage 1: Uncertainty-Based Candidate Pooling. For each unqueried node $v \in V_{\text{sub}} \setminus Q^{(\gamma-1)}$, we compute a composite uncertainty score from the output distribution p_v of $f_s^{(\gamma-1)}$ (where $p_{v,c}$

denotes the surrogate's softmax probability for class c):

$$U(v) \;=\; w_{\text{ent}}\Big(-\sum_{c=1}^{C} p_{v,c}\log p_{v,c}\Big) \;+\; w_{\text{mar}}\Big(1-\big(p_v^{(1)}-p_v^{(2)}\big)\Big). \tag{4}$$

The weights are non-negative ($w_{\text{ent}}, w_{\text{mar}} \geq 0$) and normalized ($w_{\text{ent}} + w_{\text{mar}} = 1$). The terms $p_v^{(1)}$ and $p_v^{(2)}$ are the two largest probabilities in the vector p_v. Since the margin $p_v^{(1)} - p_v^{(2)} \in [0,1]$, the second term also lies in $[0,1]$, maintaining dimensional consistency with the entropy score. We then form the candidate pool P_γ by taking the top-m_γ nodes with highest $U(v)$, where $m_\gamma = \kappa q$ for some integer $\kappa \geq 1$.

Stage 2: Diversity-Enforced Final Selection. To avoid redundant queries and leverage the model's evolving knowledge, we enforce diversity within P_γ via a k-center objective in the dynamic embedding space of the current surrogate model. Let $h_v^{(\gamma-1)}$ be the embedding of node v produced by $f_s^{(\gamma-1)}$. We select a size-q batch Q_γ via

$$Q_\gamma^* \;=\; \arg\min_{\substack{Q\subseteq P_\gamma \\ |Q|=q}} \max_{v\in P_\gamma} \min_{u\in Q} d\big(h_v^{(\gamma-1)},\, h_u^{(\gamma-1)}\big), \tag{5}$$

which is approximated by a farthest-first greedy procedure. This provides a 2-approximation to the optimal solution (see Appendix A).

In addition, we apply a per-class quota to reduce selection bias towards a single predicted class. Let the surrogate's predicted class for a node u be $\hat{y}_u = \arg\max_c p_{u,c}$. The per-class cap q_c is:

$$q_c \;=\; \max\Big(1,\; \Big\lceil \beta\cdot\frac{q}{C}\Big\rceil\Big), \qquad \forall\, c\in\{1,\dots,C\}, \quad \beta\in(0,1]. \tag{6}$$

During the greedy selection, a candidate is added to Q_γ only if the number of nodes selected for its predicted class $\hat{y}_u$ has not yet reached the cap. If the class-cap constraint makes the selection infeasible (i.e., $|Q_\gamma| < q$ after one pass), we gradually relax it by increasing β toward 1 and, if needed, lifting the cap for the least represented predicted classes until $|Q_\gamma| = q$.

3.4 Surrogate Model Training and Regularization

Training a robust surrogate from a sparse set of hard labels requires a carefully designed objective function. At each round γ, the model is trained by minimizing the composite loss:

$$\mathcal{L}_{\text{train}}(\theta_s) = \mathcal{L}_{\text{CE}}(Q^{(\gamma)}, Y_Q^{(\gamma)}) + \lambda_{\text{lap}}(\gamma)\mathcal{L}_{\text{lap}}. \tag{7}$$

Let $z_v \in \mathbb{R}^C$ denote the logits of node v from the current surrogate, and $p_v = \text{softmax}(z_v)$. All logarithms are natural. The first term is the cross-entropy loss:

$$\mathcal{L}_{\text{CE}}(Q^{(\gamma)}, Y_Q^{(\gamma)}) = -\frac{1}{|Q^{(\gamma)}|}\sum_{v\in Q^{(\gamma)}}\sum_{c=1}^{C}\mathbf{1}\big(y_v^{\text{victim}}=c\big)\log(p_{v,c}). \tag{8}$$

The second term, $\mathcal{L}_{\text{lap}}$, is a node-level adaptive Graph Laplacian regularizer:

$$\mathcal{L}_{\text{lap}} = \frac{1}{|E_{\text{sub}}|}\sum_{(i,j)\in E_{\text{sub}}}\sqrt{w_i w_j}\left\|\frac{z_i}{\sqrt{\deg(i)+1}} - \frac{z_j}{\sqrt{\deg(j)+1}}\right\|_2^2. \tag{9}$$

In practice, if the subgraph is edgeless ($|E_{\text{sub}}| = 0$), we define $\mathcal{L}_{\text{lap}} = 0$. For directed graphs, this can be adapted by using a symmetrized adjacency matrix. The components of Eq. equation 9 are defined as follows. The term $z_i \in \mathbb{R}^C$ is the unnormalized logits for node i. The node degree is $\deg(i) = |\{j : (i,j) \in E_{\text{sub}}\}|$. The node weights w_v combine normalized degree and clustering coefficients:

$$w_v = \text{clip}_{[0,1]}\Big(\alpha_{\text{node}}\,\text{normdeg}(v) + (1-\alpha_{\text{node}})\,\text{normclust}(v)\Big), \quad \alpha_{\text{node}}\in[0,1]. \tag{10}$$

Here, $cc(v)$ is the local clustering coefficient of node v in G_{sub}, and $\text{clip}_{[0,1]}(x) = \min(1, \max(0, x))$. The normalization functions $\text{normdeg}(v)$ and $\text{normclust}(v)$ apply min-max scaling to node degrees and local clustering coefficients over all nodes in V_{sub}. For min–max normalization, if the denominator is zero, we set the normalized value to 0. The regularization strength $\lambda_{\text{lap}}(\gamma)$ is scheduled heuristically as described in Appendix A.

3.5 Fine-Tuning with Self-Training

After the total query budget B is exhausted (at round Γ), we perform a final fine-tuning step via self-training. We use the final surrogate $f_s^{(\Gamma)}$ to generate predictions on all unqueried nodes. A pseudo-labeled set V_{pseudo} is curated by selecting nodes whose maximum prediction probability exceeds a hyperparameter threshold $\tau^\star \in (0.5, 1)$:

$$V_{\text{pseudo}} = \left\{ v \in V_{\text{sub}} \setminus Q^{(\Gamma)} : \max_c p_{v,c} \geq \tau^\star \right\}. \tag{11}$$

For $v \in V_{\text{pseudo}}$, we define the pseudo-label $y_v^{\text{pseudo}} = \arg\max_c p_{v,c}$ and collect $Y_{\text{pseudo}} = \{y_v^{\text{pseudo}} : v \in V_{\text{pseudo}}\}$. The model is then fine-tuned by minimizing the following loss, with a hyperparameter weight $\lambda_{\text{pseudo}} \geq 0$:

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{CE}}(Q^{(\Gamma)}, Y_Q^{(\Gamma)}) + \lambda_{\text{pseudo}}\, \mathcal{L}_{\text{CE}}(V_{\text{pseudo}}, Y_{\text{pseudo}}). \tag{12}$$

This step uses confident pseudo-labels to refine the decision boundaries without additional queries.

4 Experimental Evaluation

In this section, we conduct a series of rigorous experiments to systematically evaluate our proposed model extraction framework. Our evaluation is designed to address three central research questions: **RQ1:** What is the influence of varying query budgets on our model's effectiveness compared to baselines, given the same limited initial information? **RQ2:** How does the amount of initial information available to the attacker, specifically the size of the query pool, impact the performance of the models? **RQ3:** What are the individual contributions and effectiveness of our framework's core components? To answer these questions, we first detail our comprehensive experimental setup.

4.1 Experimental Setup

Datasets and Attack Scenario. We experiment on five diverse node classification benchmarks: co-authorship networks Coauthor-CS (CoCS) and Coauthor-Physics (CoP); co-purchase networks Amazon-Computer (AmzC) and Amazon-Photo (AmzP); and the citation network Cora-Full (Cora). To simulate a realistic black-box attack, we partition nodes into a 60% set to train the victim model f_v and a 40% global test set, which remains hidden from the attacker. The attacker can query from a pool of nodes drawn from the victim's training set. For our primary analysis on query budgets, this pool is fixed at 10% of the total nodes; to study the impact of prior information, we vary its size from 1% to 10%.

Victim Model Training Protocol. To ensure reproducibility and fairness, we pre-train the victim model and freeze its parameters for all experiments. The victim model is a Graph Convolutional Network (GCN) with hidden dimension 16 and dropout $p = 0.5$, trained on the exclusive 60% training partition. Training is performed for $T_{\text{vic}} = 1000$ epochs using the Adam optimizer with learning rate $\eta_{\text{vic}} = 1 \times 10^{-3}$ and weight decay $\lambda_{\text{wd}} = 5 \times 10^{-4}$. These settings ensure consistent with the standard GCN practice (Kipf & Welling, 2017).

Evaluation Scheme. We evaluate performance under multiple total query budgets $B \in \{5C, 10C, 15C, 20C\}$, where C is the number of classes. Queries are performed iteratively in batches of size q, leading to a total of $\Gamma = \lceil (B - |Q_0|)/q \rceil$ active learning rounds. We adopt two complementary metrics. Accuracy measures surrogate model correctness on the global test set and reflects practical utility. Fidelity quantifies agreement between surrogate and victim predictions on the test set, as defined in Eq. equation 2, and serves as the core metric for model stealing success (Jagielski et al., 2020; Podhajski et al., 2024; Wu et al., 2023). To ensure statistical significance, all experiments are repeated five times with different random seeds, and we report the mean and standard deviation of the results.

Baselines. We benchmark against a spectrum of representative methods. The **Random** baseline selects nodes uniformly at random from the query pool, serving as a lower bound. **AGE (Active Graph Embedding)** is a classic active learning approach that prioritizes nodes expected to maximally reduce uncertainty or improve embeddings (Cai et al., 2017a). **CEGA (Cost-Efficient Graph Acquisition)** is a state-of-the-art framework that selects informative nodes by jointly considering

representativeness, uncertainty, and diversity (Wang et al., 2025). Finally, the **Realistic Attack** method incorporates an edge prediction module that adds potential edges within the subgraph to enhance connectivity based on feature similarity (Guan et al., 2024). To ensure a fair comparison, all baselines operate under the same minimal information constraints as our framework.

A comprehensive breakdown of our methodology, including all model hyperparameters and complete experimental results, is presented in Appendix E.

4.2 Performance under Varying Query Budgets

To answer RQ1, we evaluate MIME against the baselines under four distinct total query budgets: 5C, 10C, 15C, and 20C, where C is the number of classes. The comprehensive Accuracy and Fidelity scores for all methods across the five benchmark datasets are detailed in Table 1.

Consistent High Performance. The results highlight a key trend: while all methods benefit from larger query budgets, MIME's performance trajectory is notably steeper and more consistent across both accuracy and fidelity. Although not always the top performer in every isolated scenario, MIME establishes a significant and robust overall advantage. This performance gap widens as the query budget increases, underscoring the framework's superior query efficiency and its ability to continuously extract value from new information.

Impressive Efficiency at Low Budgets. A key strength of MIME is its remarkable efficiency in low-budget regimes, which are critical for stealthy attacks. The framework consistently achieves high performance with a highly restricted number of queries. Most notably, in several instances, MIME's effectiveness with a minimal budget meets or even surpasses what baseline methods achieve only at their maximum budget. This demonstrates a substantial improvement in query efficiency, directly attributable to its robust cold start initialization and strategic node selection. Consequently, MIME proves its viability as a real-world attack vector, capable of replicating a target model with high fidelity at a fraction of the expected cost.

Table 1: Comparison of accuracy and fidelity across methods and total query budgets on five datasets. The best performance for each budget and dataset is highlighted in bold.

Dataset	Method	Accuracy Query Budget				Fidelity Query Budget			
		5C	10C	15C	20C	5C	10C	15C	20C
CoCS	Random	0.7262 ± 0.03	0.8103 ± 0.02	0.8396 ± 0.01	0.8565 ± 0.01	0.7429 ± 0.03	0.8327 ± 0.02	0.8642 ± 0.01	0.8846 ± 0.01
	AGE	0.7576 ± 0.03	0.8396 ± 0.02	0.8617 ± 0.01	0.8744 ± 0.01	0.7741 ± 0.03	0.8634 ± 0.02	0.8897 ± 0.01	0.9046 ± 0.01
	CEGA	0.7212 ± 0.06	0.8419 ± 0.01	0.8682 ± 0.00	0.8795 ± 0.00	0.7336 ± 0.07	0.8659 ± 0.01	0.8962 ± 0.00	0.9117 ± 0.00
	REALISTIC	0.7624 ± 0.02	0.8280 ± 0.01	0.8557 ± 0.01	0.8688 ± 0.01	0.7824 ± 0.02	0.8525 ± 0.01	0.8854 ± 0.01	0.9000 ± 0.01
	MIME	$\mathbf{0.8005 \pm 0.02}$	$\mathbf{0.8913 \pm 0.01}$	$\mathbf{0.9049 \pm 0.01}$	$\mathbf{0.9059 \pm 0.00}$	$\mathbf{0.8203 \pm 0.03}$	$\mathbf{0.9200 \pm 0.01}$	$\mathbf{0.9371 \pm 0.01}$	$\mathbf{0.9363 \pm 0.01}$
CoP	Random	0.6779 ± 0.08	0.7870 ± 0.05	0.8434 ± 0.03	0.8724 ± 0.01	0.6846 ± 0.08	0.7970 ± 0.05	0.8559 ± 0.03	0.8868 ± 0.01
	AGE	0.7963 ± 0.04	0.8336 ± 0.05	0.8865 ± 0.04	0.8927 ± 0.02	0.8053 ± 0.04	0.8457 ± 0.05	0.9027 ± 0.04	0.9087 ± 0.02
	CEGA	0.6744 ± 0.06	0.8090 ± 0.04	0.8512 ± 0.04	0.9046 ± 0.01	0.6897 ± 0.06	0.8199 ± 0.04	0.8537 ± 0.04	0.9220 ± 0.01
	REALISTIC	0.6863 ± 0.08	0.7946 ± 0.05	0.8671 ± 0.02	0.8854 ± 0.01	0.6930 ± 0.09	0.8050 ± 0.05	0.8807 ± 0.02	0.9019 ± 0.01
	MIME	$\mathbf{0.8347 \pm 0.04}$	$\mathbf{0.9228 \pm 0.01}$	$\mathbf{0.9296 \pm 0.00}$	$\mathbf{0.9331 \pm 0.00}$	$\mathbf{0.8461 \pm 0.04}$	$\mathbf{0.9407 \pm 0.01}$	$\mathbf{0.9483 \pm 0.00}$	$\mathbf{0.9523 \pm 0.00}$
AmzC	Random	0.5978 ± 0.07	0.6957 ± 0.02	0.7343 ± 0.01	0.7566 ± 0.02	0.6310 ± 0.08	0.7361 ± 0.02	0.7788 ± 0.01	0.8046 ± 0.02
	AGE	0.5781 ± 0.06	0.6863 ± 0.03	0.7295 ± 0.02	0.7598 ± 0.02	0.6100 ± 0.06	0.7278 ± 0.03	0.7726 ± 0.02	0.8068 ± 0.03
	CEGA	0.4394 ± 0.01	0.6686 ± 0.02	0.7292 ± 0.02	0.7802 ± 0.01	0.4545 ± 0.01	0.7068 ± 0.03	0.7731 ± 0.03	0.8301 ± 0.01
	REALISTIC	$\mathbf{0.6660 \pm 0.05}$	0.7067 ± 0.02	0.7491 ± 0.00	0.7631 ± 0.02	$\mathbf{0.7024 \pm 0.06}$	0.7440 ± 0.03	0.7858 ± 0.00	0.8058 ± 0.02
	MIME	0.4795 ± 0.07	$\mathbf{0.7242 \pm 0.03}$	$\mathbf{0.7974 \pm 0.02}$	$\mathbf{0.7989 \pm 0.04}$	0.5026 ± 0.07	$\mathbf{0.7624 \pm 0.04}$	$\mathbf{0.8488 \pm 0.02}$	$\mathbf{0.8487 \pm 0.05}$
AmzP	Random	0.6694 ± 0.03	0.7629 ± 0.03	0.8155 ± 0.02	0.8389 ± 0.01	0.6902 ± 0.03	0.7875 ± 0.03	0.8442 ± 0.02	0.8693 ± 0.02
	AGE	0.5152 ± 0.07	0.7820 ± 0.04	0.8240 ± 0.03	0.8363 ± 0.02	0.5293 ± 0.08	0.8056 ± 0.05	0.8508 ± 0.03	0.8631 ± 0.02
	CEGA	0.4934 ± 0.13	0.7492 ± 0.06	0.8383 ± 0.01	0.8519 ± 0.01	0.5034 ± 0.13	0.7676 ± 0.07	0.8658 ± 0.01	0.8790 ± 0.01
	REALISTIC	$\mathbf{0.7642 \pm 0.04}$	$\mathbf{0.8052 \pm 0.05}$	$\mathbf{0.8431 \pm 0.02}$	0.8545 ± 0.02	$\mathbf{0.7895 \pm 0.04}$	$\mathbf{0.8350 \pm 0.05}$	$\mathbf{0.8730 \pm 0.02}$	0.8857 ± 0.02
	MIME	0.4961 ± 0.16	0.6974 ± 0.11	0.8242 ± 0.09	$\mathbf{0.8857 \pm 0.02}$	0.5071 ± 0.18	0.7143 ± 0.12	0.8457 ± 0.09	$\mathbf{0.9146 \pm 0.02}$
Cora	Random	0.3175 ± 0.03	0.4220 ± 0.01	0.4773 ± 0.00	0.5068 ± 0.00	0.3778 ± 0.04	0.5104 ± 0.01	0.5833 ± 0.01	0.6234 ± 0.00
	AGE	0.2761 ± 0.01	0.3508 ± 0.01	0.4190 ± 0.00	0.4755 ± 0.00	0.3265 ± 0.01	0.4198 ± 0.01	0.5039 ± 0.01	0.5786 ± 0.00
	CEGA	0.2764 ± 0.01	0.3640 ± 0.01	0.4327 ± 0.01	0.4771 ± 0.00	0.3235 ± 0.01	0.4540 ± 0.01	0.5206 ± 0.01	0.5804 ± 0.00
	REALISTIC	0.3033 ± 0.01	0.4055 ± 0.00	0.4653 ± 0.01	0.4932 ± 0.01	0.3623 ± 0.02	0.4923 ± 0.01	0.5696 ± 0.01	0.6065 ± 0.01
	MIME	$\mathbf{0.3605 \pm 0.01}$	$\mathbf{0.4608 \pm 0.00}$	$\mathbf{0.5219 \pm 0.01}$	$\mathbf{0.5547 \pm 0.00}$	$\mathbf{0.4258 \pm 0.01}$	$\mathbf{0.5559 \pm 0.01}$	$\mathbf{0.6322 \pm 0.01}$	$\mathbf{0.6755 \pm 0.01}$

4.3 Performance under Varying Prior Sizes

To answer RQ2, we evaluate the performance of MIME under different levels of prior knowledge, with the initial pool size set to 1%, 3%, 5%, 8%, and 10%. For brevity, Figure 2 shows a representative subset of these results, plotting Accuracy across three benchmark datasets (CoP, AmzP, and

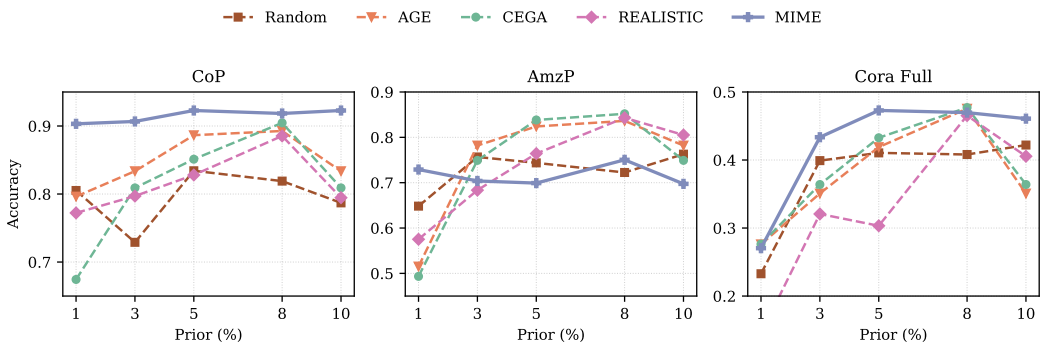

Figure 2: Accuracy of different methods under varying prior sizes (1–10%) across three representative benchmark datasets (CoP, AmzP, and Cora). The results show that MIME consistently outperforms baselines, demonstrating strength at both the lowest and highest prior knowledge levels.

Cora). The complete results for both Accuracy and Fidelity across all five datasets are provided in Appendix Figure 4.

Robustness to Prior Knowledge. Increasing the prior size does not guarantee monotonic performance gains. While MIME is the top overall performer, it shows minor fluctuations, likely due to dataset-specific traits. Crucially, these fluctuations are often smaller than those of the baselines, indicating greater stability. MIME's advantage is most pronounced at the extremes: with very small priors (1-3%), where it excels under tight constraints, and at the largest prior size (10%), where it surpasses most competitors. These results confirm that MIME is both efficient and robust, delivering reliable performance regardless of the amount of prior knowledge.

4.4 Analysis of Framework Components

To answer RQ3 and quantify each component's contribution, we conduct an ablation study using a leave-one-out methodology. This approach assesses the marginal impact of removing a single module from the full MIME framework. We evaluate six ablated configurations by individually removing: (i) DGI pre-training, (ii) diversity-based query selection, (iii) the class-balancing quota, (iv) the graph Laplacian regularizer, (v) the final self-training phase, and (vi) the use of dynamic embeddings (reverting to static ones). The complete results and detailed configuration descriptions are available in the Appendix Table 5.

Component Necessity and Synergy. The results in Table 2 confirm that each module is integral to MIME's performance. Removing foundational components like DGI pre-training, the Laplacian regularizer, the class quota, and self-training consistently degrades performance, validating their necessity. Likewise, dynamic embeddings prove superior to static ones. Despite minor dataset-specific exceptions (e.g., on AmzC), the complete framework is superior on most benchmarks, demonstrating the robustness of its integrated design. Full results for both Accuracy and Fidelity are available in Appendix Table 6.

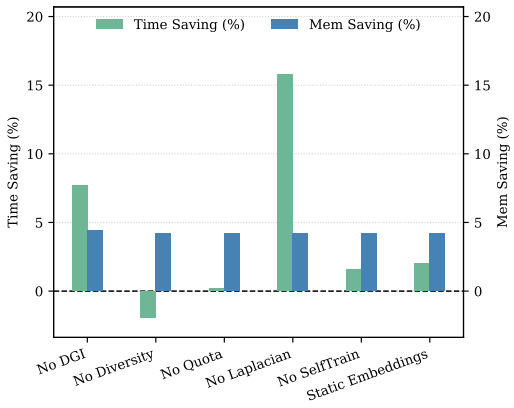

Figure 3: Comparison of computational savings from removing individual MIME components. Savings are relative to the full MIME, which operates at 14.96 ms/epoch and 7754 MB/epoch. The analysis reveals that the Laplacian regularizer is the primary source of time overhead, whereas all components have a minimal and consistent impact on memory usage.

Table 2: Ablation study results on accuracy across five benchmark datasets. The best performance for each dataset is highlighted in bold.

Ablation	CoCS	CoP	AmzC	AmzP	Cora
MIME	$\mathbf{0.9059 \pm 0.00}$	$\mathbf{0.9331 \pm 0.00}$	0.7989 ± 0.04	$\mathbf{0.8857 \pm 0.02}$	$\mathbf{0.5547 \pm 0.00}$
No DGI	0.8989 ± 0.01	0.9317 ± 0.00	0.7894 ± 0.00	0.8697 ± 0.01	0.5513 ± 0.00
No Diversity	0.9047 ± 0.00	0.9285 ± 0.00	$\mathbf{0.8128 \pm 0.01}$	0.8668 ± 0.00	0.5505 ± 0.00
No Quota	0.8970 ± 0.00	0.9238 ± 0.01	0.7799 ± 0.02	0.8773 ± 0.01	0.5517 ± 0.00
No Laplacian	0.9005 ± 0.01	0.9282 ± 0.00	0.7956 ± 0.00	0.8609 ± 0.01	0.5500 ± 0.00
No SelfTrain	0.8935 ± 0.00	0.9213 ± 0.01	0.7614 ± 0.02	0.8514 ± 0.01	0.5448 ± 0.00
Static Embeddings	0.9047 ± 0.00	0.9228 ± 0.00	0.8054 ± 0.00	0.8359 ± 0.01	0.5492 ± 0.00

Computational Efficiency. Figure 3 analyzes the cost-benefit trade-off of each component. The full MIME framework is highly efficient, as most modules like the class quota and self-training are exceptionally lightweight, adding negligible time cost. The only notable overhead comes from the Laplacian regularizer, but its accuracy benefits justify this trade-off. Crucially, all ablated versions have nearly identical memory footprints, with the removal of any single component yielding a memory saving of only about 4.0-4.4%. This confirms that MIME's components achieve performance gains with minimal computational overhead.

5 Related Work

GNN Model Extraction and Unrealistic Assumptions. Prior work on GNN model extraction often relies on unrealistic assumptions that limit practical application. Many methods assume access to a large, in-distribution dataset (Wu et al., 2022b; 2021; 2022a), soft-label feedback like probabilities instead of hard labels (Carlini et al., 2020), or unconstrained query budgets that ignore real-world costs (Yang et al., 2023). Even recent graph-specific active learning frameworks such as AGE, GRAIN, and CEGA require an initial set of labeled seed nodes (Cai et al., 2017b; Zhang et al., 2021; Wang et al., 2025). These simplifying assumptions prevent existing work from addressing the fully constrained environments faced by real-world adversaries.

Enabling Techniques for "Cold Start in the Dark." Recent work has started to address this realism gap (Guan et al., 2024; Zhuang et al., 2024a), yet these methods often assume partial supervision and do not offer a complete cold start solution. Two key technologies are relevant: unsupervised graph representation learning methods can create high-quality, label-free embeddings from structure alone (Velickovic et al., 2019; You et al., 2020; Hassani & Khasahmadi, 2020), while active learning frameworks like AGE, GRAIN, and CEGA can optimize query budgets but still require initial labeled seeds to function. MIME's key contribution is to bridge this gap. It is one of the first to use unsupervised embeddings as a true zero-label initializer for a budget-aware, hard-label active learning loop. This enables effective extraction under realistic constraints, setting our work apart.

6 Conclusion

In this paper, we introduced MIME, a novel framework for practical GNN model extraction. We formally defined and addressed the stringent "Cold Start in the Dark" problem, a realistic scenario characterized by minimal initial information, a tight query budget, and hard-label feedback. By integrating unsupervised pre-training with a query-efficient active learning loop, MIME effectively overcomes these challenges. Our experiments confirmed its superiority, showing that it consistently outperforms state-of-the-art baselines in both Accuracy and Fidelity.

MIME's success closes the gap between theoretical and practical GNN extraction attacks, validating a potent, low-barrier attack vector by showing how adversaries can leverage graph topology before any labels are queried. This finding proves that defenses must evolve beyond simple query monitoring. Looking ahead, crucial future work involves not only extending the MIME framework to the challenging inductive setting but also developing a new class of robust countermeasures capable of detecting the subtle statistical footprints of such stealthy attacks.

Ethics Statement

We acknowledge that this work details a model extraction attack, a methodology with potential for malicious use. Our primary motivation is defensive: by developing and analyzing a more realistic and efficient attack vector, we aim to highlight critical vulnerabilities in current GNN-based MLaaS platforms. This research is intended to serve as a benchmark for the security community, enabling the development and evaluation of more robust defense mechanisms against such threats. The described methods were developed in a controlled, simulated environment. We believe that transparently discussing these vulnerabilities is crucial for motivating and informing the creation of stronger security protocols. Our work adheres to the principles of responsible research by focusing on the security implications and providing insights for defenders.

Reproducibility Statement

We have made every effort to ensure the reproducibility of our results. A detailed description of our proposed framework, MIME, including the unsupervised pre-training, iterative query strategy, and surrogate model training, is provided in Section 3. The full algorithm is presented in Algorithm 1 in the appendix. Our complete experimental setup, including dataset descriptions, data partitioning, and victim model training protocols, is detailed in Section 4.1. The specific hyperparameters used for all experiments are also listed in the appendix. All datasets used (CoCS, CoP, AmzC, AmzP, and Cora) are publicly available benchmarks. To facilitate full reproduction of our findings, we provide our source code, including the implementation of our method and all baselines, in the supplementary materials.

References

Hongyun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. Active graph embedding for node classification. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 639–648, 2017a.

Hongyun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. Active learning for graph neural networks via uncertainty and diversity. In *2017 IEEE International Conference on Data Mining (ICDM)*, pp. 1151–1156, 2017b.

Nicholas Carlini, Matthew Jagielski, and Ilya Mironov. Cryptanalytic extraction of neural network models. In *Annual International Cryptology Conference*, pp. 189–218, 2020.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models revisited. In *30th USENIX Security Symposium (USENIX Security 21)*, pp. 2633–2650, 2021.

Varun Chandrasekaran, Kamalika Chaudhuri, Irene Giacomelli, Somesh Jha, and Songbai Yan. Beyond oracles: The future of model extraction attacks. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 1309–1326, 2020.

Zhan Cheng, Bolin Shen, Tianming Sha, Yuan Gao, Shibo Li, and Yushun Dong. Atom: A framework for detecting query-based model extraction attacks for graph neural networks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, volume 2, pp. 322–333, 2025.

Anuj Dubey, Emre Karabulut, Amro Awad, and Aydin Aysu. High-fidelity model extraction attacks via remote power monitors. In *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pp. 207–210, 2022.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pp. 1263–1272, 2017.

Xirong Gong, Qi Wang, Yifeng Chen, Wei Yang, and Xiaohong Jiang. Model extraction attacks and defenses on cloud-based machine learning models. *IEEE Communications Magazine*, 58(12): 83–89, 2020. doi: 10.1109/MCOM.001.2000196.

Teofilo F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985. doi: 10.1016/0304-3975(85)90224-5.

Faqian Guan, Tianqing Zhu, Hanjin Tong, and Wanlei Zhou. A realistic model extraction attack against graph neural networks. *Knowledge-Based Systems*, 285:111657, 2024.

Shubham Gupta, Pradeep Kumar, and Amit Singh. Graph neural networks: Applications and challenges. *ACM Computing Surveys*, 54(8):1–35, 2021.

Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *International Conference on Machine Learning*, pp. 4116–4126, 2020.

Xinlei He, Jinyuan Jia, Michael Backes, Neil Zhenqiang Gong, and Yang Zhang. Stealing links from graph neural networks. In *30th USENIX Security Symposium (USENIX Security 21)*, pp. 2689–2706, 2021.

Xuanli He, Qiongkai Zhu, Yi Zeng, Lingjuan Lyu, Fangzhao Wu, Jiwei Li, and Ruoxi Jia. Textknockoff: Model extraction attacks on text generation apis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12345–12358, 2023.

Jun Hou, Jiali Qian, Yulong Wang, Xinyu Li, Hong Du, and Lei Chen. Ml defense: Against prediction api threats in cloud-based machine learning service. In *2019 IEEE/ACM 27th International Symposium on Quality of Service (IWQoS)*, pp. 1–10. IEEE, 2019. doi: 10.1145/3326285.3329042.

Matthew Jagielski, Nicholas Carlini, David Berthelot, Alex Kurakin, and Nicolas Papernot. High accuracy and high fidelity extraction of neural networks. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 1345–1362, 2020.

Manish Kesarwani, Bhaskar Mukhoty, Vijay Arya, and Sameep Mehta. Model extraction warning in mlaas paradigm. In *Proceedings of the 34th Annual Computer Security Applications Conference (ACSAC)*, pp. 371–380. ACM, 2018. doi: 10.1145/3274694.3274740.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Kalpesh Krishna, Gaurav Singh Tomar, Ankur P. Parikh, Nicolas Papernot, and Mohit Iyyer. Greylm: Stealing language models with grey-box access. In *International Conference on Learning Representations*, 2020.

Pengcheng Li, Jinfeng Yi, and Lijun Zhang. Query-limited model extraction attacks against black-box machine learning models. In *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 1200–1205, 2018.

Alexander Muzio, Leslie O'Bray, and Karsten Borgwardt. Biological data learning: A survey of deep learning applications in computational biology. *Nature Methods*, 18(11):1234–1248, 2021.

Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4949–4958, 2019.

Marcin Podhajski, Jan Dubiński, Franziska Boenisch, Adam Dziedzic, Agnieszka Pregowska, and Tomasz Michalak. Efficient model-stealing attacks against inductive graph neural networks. In *Advances in Neural Information Processing Systems*, 2024.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

Florian Tramer, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. In *25th USENIX Security Symposium (USENIX Security 16)*, pp. 601–618, 2016.

Petar Velickovic, William Fedus, William L. Hamilton, Pietro Lio, Yoshua Bengio, and R. Devon Hjelm. Deep graph infomax. In *International Conference on Learning Representations*, 2019.

Binghui Wang and Neil Zhenqiang Gong. Shrewd attack: Model extraction under resource constraints. In *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 36–52, 2018.

Yixu Wang, Jie Li, Hong Liu, Yan Wang, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Queryfake: Defending against model extraction attacks with fake queries. In *European Conference on Computer Vision*, pp. 567–583, 2022.

Zebin Wang, Menghan Lin, Bolin Shen, Ken Anderson, Molei Liu, Tianxi Cai, and Yushun Dong. Cega: A cost-effective approach for graph-based model extraction and acquisition. *arXiv preprint arXiv:2506.17709*, 2025.

Lirong Wu, Stan Z. Li, and Steven C. H. Hoi. Graph neural networks for anomaly detection: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 33(8):3663–3680, 2022a.

Shiwen Wu, Fuzhen Sun, Wentao Zhang, Xin Xie, and Bin Cui. Link prediction with graph neural networks: A survey. In *IEEE Transactions on Knowledge and Data Engineering*, volume 34, pp. 2137–2155, 2022b.

Yixin Wu, Xinlei He, Pascal Berrang, Mathias Humbert, Michael Backes, Neil Zhenqiang Gong, and Yang Zhang. Link stealing attacks against inductive graph neural networks. *arXiv preprint arXiv:2405.05784*, 2024.

Zhiyuan Wu, Sheng Sun, Yuwei Wang, Min Liu, Ke Xu, Wen Wang, Xuefeng Jiang, Bo Gao, and Jinda Lu. Model extraction attacks on graph neural networks: A comprehensive survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7892–7910, 2023.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2021.

Panpan Yang, Qinglong Wu, and Xinming Zhang. Efficient model extraction by data set stealing, balancing, and filtering. *IEEE Internet of Things Journal*, 2023. doi: 10.1109/JIOT.2023.3293904.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5812–5823, 2020.

Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V. Chawla. Grain: Graph active learning with influence maximization. In *Proceedings of the Web Conference 2021*, pp. 1613–1622, 2021.

Wenbin Zhang, Wei Chen, Tongliang Liu, and Qiang Yang. Batch active learning for graph neural networks via uncertainty and diversity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8623–8631, 2022.

Jiawei Zhuang, Qi Zhang, and Chuxu Zhang. Stealing graph neural networks: Methods and defenses. In *Proceedings of the ACM Web Conference 2024*, pp. 1234–1245, 2024a.

Jiawei Zhuang, Qi Zhang, and Chuxu Zhang. Unveiling the secrets without data: Can graph neural networks be exploited through data-free model extraction attacks? In *31st USENIX Security Symposium (USENIX Security 24)*, 2024b.

APPENDIX

A SUPPLEMENTARY INSTRUCTIONS FOR FORMULAS

K-center Objective Details The k-center objective seeks to select a subset of points Q from a larger set P_γ that minimizes the maximum distance from any point in P_γ to its nearest point in Q. This is formally expressed as:

$$\phi(Q) = \max_{v \in P_\gamma} \min_{u \in Q} d\big(h_v^{(\gamma-1)}, h_u^{(\gamma-1)}\big), \tag{13}$$

where $d(\cdot, \cdot)$ is the distance metric. The farthest-first greedy algorithm provides a 2-approximation for this NP-hard problem.

Laplacian Regularization Schedule: Intuition The regularization strength $\lambda_{\text{lap}}(\gamma)$ is not static. An effective schedule starts with a small λ_{lap} when the number of queried labels is low to avoid over-smoothing based on a potentially biased initial surrogate model. As more labels are acquired and the model becomes more confident, λ_{lap} can be increased to enforce greater smoothness and improve generalization. For instance, a simple schedule could be $\lambda_{\text{lap}}(\gamma) = \lambda_0 \cdot \min(1, |Q^{(\gamma)}|/B_{thresh})$, where λ_0 is the base regularization strength and B_{thresh} is a budget threshold after which the regularization is fully active. This adaptive approach prevents early, aggressive regularization from washing out the learning signal from the few available labels.

B NOTATION TABLE

Symbol	Description
$G_{\text{full}} = (V_{\text{full}}, E_{\text{full}})$	Full (hidden) graph of the victim model
$G_{\text{sub}} = (V_{\text{sub}}, E_{\text{sub}})$	Induced subgraph observed by the attacker
N, E	Number of nodes and edges in G_{sub}, i.e., $N = \lvert V_{\text{sub}}\rvert, E = \lvert E_{\text{sub}}\rvert$
$A_{\text{sub}}, X_{\text{sub}}$	Adjacency and feature matrix on G_{sub}
d, d_e	Input feature dimension and dynamic embedding dimension
C	Number of classes
$cc(v)$	Local clustering coefficient of node v
f_v, f_s	Victim and surrogate GNNs; $f_v(v) \in \mathbb{R}^C$ is victim's output vector
y_v^{victim}	Hard-label returned by the API: $y_v^{\text{victim}} = \arg\max_c f_v(v)_c$
B	Total query budget (issued in batches)
q	Batch size per round (also $\lvert Q_0\rvert$ if $B \geq q$)
Γ	Number of rounds: $\Gamma = \lceil (B - \lvert Q_0\rvert)/q \rceil$
$H^{(0)} = \{h_v^{(0)}\}$	DGI pre-trained (static) embeddings (cold start only)
$h_v^{(\gamma)}$	Round-γ embedding from $f_s^{(\gamma)}$ (dynamic)
$Q^{(\gamma)}$	Queried set up to round γ; $Q^{(0)} = Q_0$
κ	Pool factor in candidate size $m_\gamma = \kappa q$
β	Per-class cap factor (Eq. equation 6)
$U(v)$	Uncertainty score (entropy+margin) in Eq. equation 4
$d(\cdot, \cdot)$	Angular distance (cosine-induced metric) used in $k - center$
z_v, p_v	Logits and softmax probability for node v
λ_{lap}	Laplacian regularization weight (Eq. equation 7)
$\tau^\star$	Confidence threshold for pseudo-labels in self-training
λ_{pseudo}	Weight for pseudo-label loss (Eq. equation 12)
$T_{\text{vic}}, T_{\text{DGI}}, T_{\text{tr}}$	Victim training, DGI pre-training, and per-round training epochs

C PROOFS AND JUSTIFICATIONS

C.1 *Necessity of DGI at Cold Start*

We recall the DGI objective from Eq. equation 3, which maximizes a mutual information bound between local embeddings h_v and the global summary s (Velickovic et al., 2019). This ensures $H^{(0)}$ encodes structural patterns (communities, roles) without labels. The initial query set Q_0 is chosen by farthest-first k-center (Eq. equation 13), which approximates the minimum covering radius within a factor 2 (Gonzalez, 1985). Thus, Q_0 provides diverse, label-free coverage—strictly better than random seeds. *Note:* As stated in §3.2, $H^{(0)}$ is used only at cold start.

C.2 *Farthest-First (K-Center) for Diversity*

We recall the diversity objective in Eq. equation 5, defined over the candidate pool P_γ. Using the angular distance $d(\cdot,\cdot)$ (see §3.2), the greedy farthest-first heuristic achieves $\phi(Q_{\mathrm{FF}}) \le 2\phi(Q^*)$ (Gonzalez, 1985). This ensures constant-factor coverage, unlike k-means, which lacks worst-case guarantees.

C.3 *Dynamic Embeddings for Diversity*

We recall the uncertainty score $U(v)$ from Eq. equation 4. After uncertainty filtering, diversity is applied in the *dynamic embedding space* $\{h_v^{(\gamma-1)}\}$ produced by $f_s^{(\gamma-1)}$. Because embeddings evolve as more labels are queried, this keeps diversity aligned with the surrogate's current decision geometry.

C.4 *Necessity of Adaptive Laplacian Regularization*

We recall the training objective in Eq. equation 7, where $\mathcal{L}_{\mathrm{lap}}$ (Eq. equation 9) penalizes discrepancies across edges with node-adaptive weights (Eq. equation 10). A spectral view shows that regularization attenuates high-frequency modes, but excessive smoothing on sparse/heterophilous subgraphs is harmful. Hence λ_{lap} is scheduled adaptively (Appendix A), gated by subgraph connectivity, ramped with labeled fraction, and modulated by homophily/spectral cues.

C.5 *Convergence Trend*

Accuracy and Fidelity are the main evaluation metrics (Eq. equation 2); for intuition, define a surrogate–victim risk $R_\gamma = \mathbb{E}[\mathbf{1}\{\arg\max f_s^{(\gamma)} \neq \arg\max f_v\}]$. Uncertainty sampling reduces error-prone regions; diversity prevents redundancy. Thus, R_γ decreases monotonically in trend and is bounded below by 0, consistent with observed empirical convergence.

C.6 *Computational Complexity*

We recall the round structure and symbols from §3.3. DGI pre-training costs $O(T_{\mathrm{DGI}}Ed)$. Each round costs $O(Ed+NC+m_\gamma q d_e+T_{\mathrm{tr}}Ed)$. The term $m_\gamma q d_e$ arises from the farthest-first selection, where $m_\gamma = \kappa q$; its complexity can be optimized in practice with data structures like heaps. Total cost:

$$O\Big(T_{\mathrm{DGI}}Ed + \tfrac{B}{q}(T_{\mathrm{tr}}Ed + NC + \kappa q^2 d_e)\Big).$$

Since $B \ll N$, training dominates. Space is $O(E + N(d + d_e + C))$, so MIME is polynomial-time feasible.

D FULL ALGORITHM SPECIFICATION

Algorithm 1: The proposed framework of MIME

Initialization: Pre-train DGI on $(A_{\text{sub}}, X_{\text{sub}})$ to obtain initial embeddings $H^{(0)}$.
Select initial nodes Q_0 by farthest-first on $H^{(0)}$ with $|Q_0| = \min(q, B)$ from V_{sub}.
Query victim API to get hard labels Y_{Q_0}, where $y_v^{\text{victim}} = \arg\max_c [f_v(A_{\text{full}}, X_{\text{full}})]_{v,c}$.
Train the initial surrogate $f_s^{(0)}$ on (Q_0, Y_{Q_0}) by minimizing
$\mathcal{L}_{\text{train}} = \mathcal{L}_{\text{CE}}(Q_0, Y_{Q_0}) + \lambda_{\text{lap}}(0)\mathcal{L}_{\text{lap}}$ for T_{tr} epochs.
for ***Cycle*** γ *from* 1 *to* $\Gamma = \lceil (B - |Q_0|)/q \rceil$ **do**
  if $|Q^{(\gamma-1)}| + q \leq B$ **then**
    Evaluate uncertainty score $U(v)$ for all $v \in V_{\text{sub}} \setminus Q^{(\gamma-1)}$ using Eq. equation 4.
    Build candidate pool P_γ with top-m_γ nodes, $m_\gamma = \kappa q$.
    Obtain dynamic embeddings $h_v^{(\gamma-1)}$ from $f_s^{(\gamma-1)}$.
    Select and query q nodes Q_γ via farthest-first on P_γ with class cap (Eq. equation 6).[a]
    Obtain victim labels Y_{Q_γ} and set $Q^{(\gamma)} = Q^{(\gamma-1)} \cup Q_\gamma$.
  else
    Set $Q^{(\gamma)} = Q^{(\gamma-1)}$.
  end
  Train $f_s^{(\gamma)}$ on $\{Q^{(\gamma)}, G_{\text{sub}}\}$ for T_{tr} epochs by minimizing Eq. equation 7.
end
Self-training: Form $V_{\text{pseudo}} = \{v : \max_c p_{v,c} \geq \tau^\star\}$ with labels Y_{pseudo} from $f_s^{(\Gamma)}$, and fine-tune with Eq. equation 12.
Return:
Queried nodes $\{Q^{(1)}, ..., Q^{(\Gamma)}\}$ and final surrogate f_s^{final}.

[a] If the remaining budget is less than q, this step selects only the number of remaining nodes allowed by the budget.

E SUPPLEMENTARY EXPERIMENTAL DETAILS

This section provides additional details regarding our experimental setup and presents supplementary results to ensure full reproducibility.

E.1 HYPERPARAMETER SETTINGS

All experiments were conducted using a fixed random seed of 42 to ensure deterministic behavior. The hyperparameters for the victim model and our proposed MIME framework were kept consistent across all datasets to ensure a fair and robust evaluation. The specific settings are detailed in Table 3 and Table 4.

Table 3: Hyperparameter settings for the victim GCN model.

Victim Model Parameter	**Value**
Architecture	GCN
Hidden Dimension	16
Dropout	0.5
Learning Rate	1×10^{-3}
Weight Decay	5×10^{-4}
Training Epochs	1000

Table 4: Hyperparameter settings for the MIME framework.

MIME Parameter	**Value**
Surrogate Hidden Dimension	128
Surrogate Dropout	0.5
Edge Dropout	0.4
Label Smoothing	0.03
Epochs per Round	100
Final Training Epochs	200
Laplacian Lambda (λ_{lap})	5×10^{-4}
Laplacian Mode	prob
K-Center Pool Multiplier (κ)	5
DGI Pre-training Epochs	500

E.2 ADDITIONAL RESULTS AND FIGURES

This subsection contains supplementary figures and tables that provide a more comprehensive view of our experimental results, including the full performance data under varying prior sizes and the detailed ablation study outcomes.

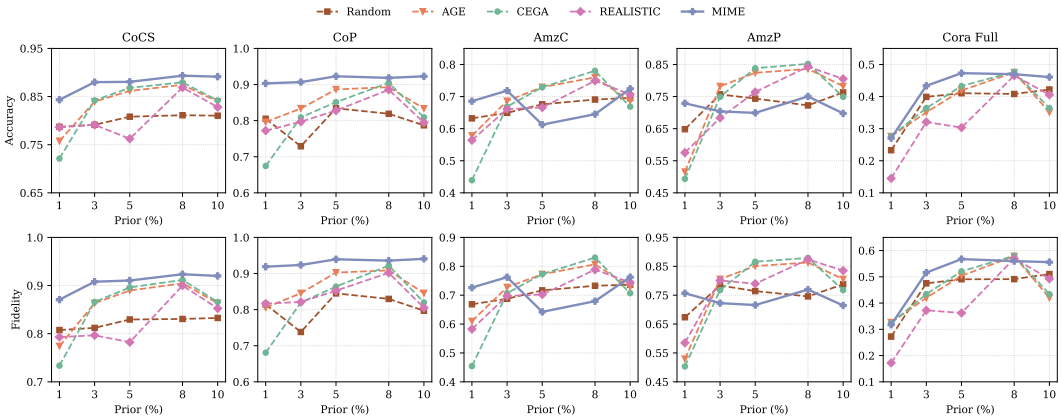

Figure 4: Complete results for accuracy and fidelity under varying prior sizes (1% to 10%) across all five benchmark datasets. These plots show the performance of MIME and all baseline methods, providing a comprehensive view of how initial information availability impacts extraction success.

Table 5: Detailed description of ablation study configurations. Each configuration disables or alters one component from the full MIME framework (referred to as the Baseline).

Configuration	**Component Modified**	**Description of Change**
no-DGI	Unsupervised Pre-training	Skips the DGI phase; the surrogate model is randomly initialized.
no-Diversity	Query Selection	Removes k-center diversity logic; selects nodes on uncertainty alone.
no-Quota	Query Selection	Disables the per-class quota for balancing query batches.
no-Laplacian	Model Training	Removes the Laplacian regularizer term from the training loss.
no-Self-Training	Fine-tuning	Omits the final self-training step after the budget is exhausted.
Static Embeddings	Query Selection	Uses static embeddings from the cold start for all diversity checks.

Table 6: Complete ablation study results on accuracy and fidelity. This table shows the performance of the full MIME framework and six ablated versions across five benchmark datasets.

Ablation	**Accuracy**					**Fidelity**				
	CoCS	**CoP**	**AmzC**	**AmzP**	**Cora**	**CoCS**	**CoP**	**AmzC**	**AmzP**	**Cora**
MIME	$\mathbf{0.9059 \pm 0.00}$	$\mathbf{0.9331 \pm 0.00}$	0.7989 ± 0.04	$\mathbf{0.8857 \pm 0.02}$	$\mathbf{0.5547 \pm 0.00}$	$\mathbf{0.9363 \pm 0.00}$	$\mathbf{0.9523 \pm 0.00}$	0.8487 ± 0.04	$\mathbf{0.9146 \pm 0.00}$	$\mathbf{0.6755 \pm 0.00}$
No DGI	0.8989 ± 0.01	0.9317 ± 0.00	0.7894 ± 0.00	0.8697 ± 0.01	0.5513 ± 0.00	0.9292 ± 0.01	0.9504 ± 0.00	0.8364 ± 0.00	0.8930 ± 0.02	0.6736 ± 0.00
No Diversity	0.9047 ± 0.00	0.9285 ± 0.00	$\mathbf{0.8128 \pm 0.01}$	0.8668 ± 0.00	0.5505 ± 0.00	0.9361 ± 0.00	0.9454 ± 0.00	0.8351 ± 0.01	0.8939 ± 0.01	0.6721 ± 0.00
No Quota	0.8970 ± 0.00	0.9238 ± 0.01	0.7799 ± 0.02	0.8773 ± 0.01	0.5517 ± 0.00	0.9295 ± 0.00	0.9425 ± 0.01	0.8289 ± 0.02	0.9073 ± 0.01	0.6717 ± 0.00
No Laplacian	0.9005 ± 0.01	0.9282 ± 0.00	0.7956 ± 0.00	0.8609 ± 0.01	0.5500 ± 0.00	0.9304 ± 0.01	0.9451 ± 0.00	0.8429 ± 0.01	0.8839 ± 0.01	0.6735 ± 0.00
No SelfTrain	0.8935 ± 0.00	0.9213 ± 0.01	0.7614 ± 0.02	0.8514 ± 0.01	0.5448 ± 0.00	0.9232 ± 0.00	0.9403 ± 0.01	0.8084 ± 0.02	0.8729 ± 0.01	0.6635 ± 0.00
Static Embeddings	0.9047 ± 0.00	0.9228 ± 0.00	0.8054 ± 0.00	0.8359 ± 0.01	0.5492 ± 0.00	0.9364 ± 0.00	0.9398 ± 0.01	$\mathbf{0.8574 \pm 0.00}$	0.8566 ± 0.01	0.6672 ± 0.00

F LLM USAGE STATEMENT

During the preparation of this manuscript, we utilized a large language model (LLM) as an assistive tool. The LLM's role was primarily focused on improving the clarity and conciseness of the text. This included rephrasing sentences and paragraphs for better readability, correcting grammar, and ensuring a consistent writing style throughout the paper. Additionally, the LLM provided assistance with LaTeX formatting, helping to structure tables, figures, and other elements in accordance with the conference template.

The core research ideas, experimental design, analysis, and conclusions presented in this paper were conceived and executed entirely by the human authors. The LLM did not contribute to the research ideation. The authors have reviewed, edited, and take full responsibility for the final content and its scientific accuracy.