Transferable Graph Condensation from the Causal Perspective

Huaming Du[1], Yijie Huang[1], Su Yao[2], Yiying Wang[3], Yueyang Zhou[3], Jingwen Yang[3], Jinshi Zhang[3], Han Ji[3], Yu Zhao[1], Guisong Liu[1], Hegui Zhang[4*], Carl Yang[5], Gang Kou[6*]

[1]Southwestern University of Finance and Economics
[2]Tsinghua University
[3]Ant Group
[4]Dongbei University of Finance and Economics
[5]Emory University
[6]Xiangjiang Laboratory
dhmfcc@swufe.edu.cn, yaosu@tsinghua.edu.cn, hegui.zhang@qq.com, kou.gang@qq.com

Abstract

The increasing scale of graph datasets has significantly improved the performance of graph representation learning methods, but it has also introduced substantial training challenges. Graph dataset condensation techniques have emerged to compress large datasets into smaller yet information-rich datasets, while maintaining similar test performance. However, these methods strictly require downstream applications to match the original dataset and task, which often fails in cross-task and cross-domain scenarios. To address these challenges, we propose a novel causal-invariance-based and transferable graph dataset condensation method, named **TGCC**, providing effective and transferable condensed datasets. Specifically, to preserve domain-invariant knowledge, we first extract domain causal-invariant features from the spatial domain of the graph using causal interventions. Then, to fully capture the structural and feature information of the original graph, we perform enhanced condensation operations. Finally, through spectral-domain enhanced contrastive learning, we inject the causal-invariant features into the condensed graph, ensuring that the compressed graph retains the causal information of the original graph. Experimental results on five public datasets and our novel **FinReport** dataset demonstrate that TGCC achieves up to a 13.41% improvement in cross-task and cross-domain complex scenarios compared to existing methods, and achieves state-of-the-art performance on 5 out of 6 datasets in the single dataset and task scenario.

Introduction

Graph Neural Networks (GNNs) (Wang et al. 2022; Guo, Liu, and Li 2023; Sun et al. 2024a; Lu et al. 2025) have garnered widespread attention for their exceptional ability to represent complex graph data and have been applied in many real-world scenarios, including social networks (Zhang et al. 2024a), enterprise risk prediction (Zhao et al. 2022), and recommender systems (Zhang et al. 2025). It is widely believed that there exists a scaling law between dataset size and the capability of deep learning models (Kaplan et al.
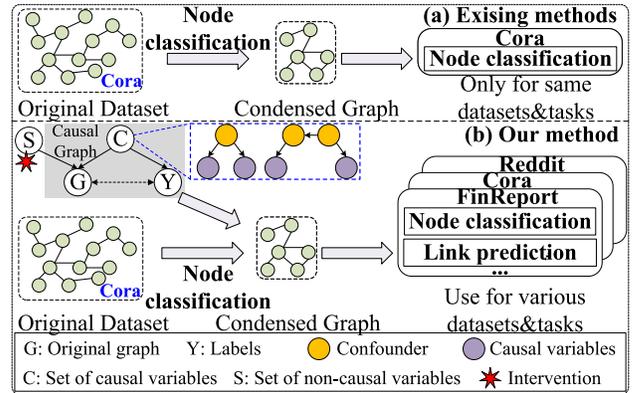

Figure 1: The pipeline of existing graph condensation methods and our TGCC method. The main differences between TGCC and existing methods lie in the extraction of causal-invariant features from the graph structure based on causal intervention to achieve transferable graph condensation.

2020). However, large-scale graph datasets also present significant challenges in terms of storage, processing, and computational resources. On one hand, specific applications such as neural architecture search (Zhang et al. 2024c), continual learning (Li et al. 2024), and pre-training (Yu et al. 2025; Liu et al. 2025) require repetitive training on these datasets, resulting in substantial computational costs. On the other hand, for users with limited computational resources, training models on large-scale graph datasets can be extremely time-consuming or even infeasible. Recently, several graph condensation (GC) methods (Sun et al. 2024b) have been proposed to alleviate these issues.

However, as shown in Fig. 1, most existing GC methods are designed based on statistical correlations to compress specific datasets, typically optimized for a single dataset and task. This limitation severely restricts the applicability of these methods in real-world scenarios, where users encounter new or custom data and tasks that differ significantly from the condensed graph datasets derived from publicly available datasets. Compared to applying GC methods

on a few public graph datasets, such scenarios are much more common and challenging. While graph transfer learning (Zhuang et al. 2020), graph meta-learning (Fang et al. 2023), and graph foundation models (Liu et al. 2025) can partially address this issue, they rely on fixed model architectures or complex training strategies, and require substantial computational resources. Therefore, a natural question arises: *Can the condensed graph dataset be used to train models that adapt to various tasks and different datasets?*

This paper focuses on the performance of models trained on condensed datasets when transferred to new datasets or tasks. Ideally, enhanced transferability would enable users to train models more efficiently, achieving better performance. However, current graph dataset condensation methods still face two major challenges.

Challenge 1: **Efficient and Fast Cross-task Adaptation**. Existing graph condensation algorithms typically condense, train, and test based on the same task, which prevents the trained models from generalizing effectively to other tasks. For example, using the Ogbn-arxiv dataset, we condense the dataset through node classification and then test the model's performance on link prediction. The performance of existing methods is on average **3.2%** lower than the ground truth (using only the simple 2-layer GCN), highlighting the difficulty of cross-task transferability.

Challenge 2: **Preservation of Causal Invariant Information**. Existing studies mainly perform condensation, training, and testing on the same dataset. The condensed graphs fail to capture the causal invariant information contained in the original graphs, leading to poor transferability of models trained on the condensed data. Using the Ogbn-arxiv as an example, we condense the dataset, then train a 2-layer GCN and generalize it to five different datasets by replacing the final layer with a new linear layer. However, the current methods still fail to outperform the simple GCN model, with an average performance loss of **9.8%**.

To address the above challenges, we propose a causal-invariance-based and transferable graph dataset condensation framework, named **TGCC**. Unlike existing methods that typically focus on condensing a specific dataset for a single task, TGCC is designed to encourage the condensed graph to retain the most universal and information-rich patterns, thereby achieving better transferability across tasks and datasets. Specifically, to enhance transferability, TGCC first extracts causal-invariant features from the graph's structure. To better capture the structural and feature information of the original graph, TGCC performs contrastive condensation operations. Finally, through spectrum-level enhanced contrastive learning, the causal-invariant features are injected into the condensed graph, ensuring that the condensed graph retains the causal information of the original graph. Additionally, we construct a novel financial graph dataset, named **FinReport**, which captures the correspondence between corporate financial reports and analyst research reports, and release it as an open-source resource for the research community.

In summary, the main contributions are stated as follows:
● We propose TGCC, a causal-invariant and transferable graph dataset condensation framework. To the best of our knowledge, TGCC is the first graph condensation method from a causal perspective that supports transferability.
● TGCC integrates spectral-domain intervention strategy and contrastive condensation strategy, leveraging contrastive learning to inject causal knowledge into the condensed graph, thereby enhancing its transferability across tasks and datasets.
● We construct a novel financial dataset, FinReport, capable of capturing the correspondence between corporate financial reports and company research reports, and we release it as an open-source resource for the research community.
● Extensive experiments on six real-world datasets demonstrate that TGCC achieves state-of-the-art performance in both single-task and cross-dataset/cross-task scenarios.

Related Works
Graph Dataset Condensation
The current GC methods are mainly divided into three categories: gradient matching, distribution matching, and trajectory matching. In recent years, several graph dataset condensation methods have been proposed to improve performance on single datasets and tasks (Xu et al. 2024; Zhang et al. 2024b; Gao et al. 2025b; Yu et al. 2025; Xiao et al. 2025). For instance, GCond (Jin et al. 2022b) first introduced the gradient matching method for graph condensation. ST-GCond (Yang et al. 2025) offers a self-supervised and transferable graph dataset condensation method to address challenges in cross-task and cross-dataset scenarios. However, most existing methods are still designed for specific datasets and tasks. The method most similar to ours, ST-GCond, which achieves transferability by pre-training multiple self-supervised model to extract fundamental information from the data. However, ST-GCond is costly and fails to extract causal-invariant features from graph data. Therefore, how to achieve transferable condensation remains an under-explored issue and requires further research.

Causal Inference in Graph Neural Networks
Causality studies explore the relationships between variables (Pearl, Glymour, and Jewell 2016), and have demonstrated many benefits in graph learning (Fan et al. 2022; Dai et al. 2024). For example, RGCL (Li et al. 2022) introduces an invariance perspective in self-supervised learning and proposes a method to preserve stable semantic information. GCIL (Mo et al. 2024) introduces spectral graph augmentation and proposes a new graph contrastive learning (GCL) method. UIL (Sui et al. 2025) provides a unified perspective on invariant graph learning, emphasizing both structural and semantic invariance principles to identify more robust stable features for graph classification. Unlike these studies, we explore a transferable graph condensation task from a causality perspective and propose a novel graph condensation framework based on causal theory.

The Proposed Model: TGCC
Overall Framework
Given a graph dataset $\mathcal{G} = (X, A, Y)$, where $X \in \mathbb{R}^{N \times d}$ represents the feature matrix, $A \in \mathbb{R}^{N \times N}$ represents the

adjacency matrix, N is the number of nodes, and d is the node feature dimension, our goal is to generate a smaller synthetic graph dataset $\mathcal{G}_s = (X_s, A_s, Y_s)$, where the number of nodes $m \ll N$. We aim to train models on the synthetic graph $\mathcal{G}_s$ that achieve similar test performance to those trained on the original graph $\mathcal{G}$ for the same task, while preserving generalization ability when transferred to new datasets or tasks. Therefore, we propose the TGCC framework as illustrated in Figure 2, which mainly consists of *three modules*: Causal Invariant Feature Extraction, Graph Contrastive Condensation, and Spectral-domain Enhanced Contrastive Learning.

Causal Invariant Feature Extraction

According to the study (Liu et al. 2022; Mo et al. 2024), perturbing the structure of the original graph alters the strength of frequency components in the graph spectrum, with the lowest-frequency information being approximately regarded as an invariant pattern across different views. Consistent with prior research (Mo et al. 2024), we consider the low-frequency components in the graph as causal content, while the high-frequency components are treated as non-causal content. Therefore, we intervene on the non-causal variable S in Figure 1 by disturbing the high-frequency information while keeping the low-frequency information unchanged.

Specifically, given the original adjacency matrix A, our goal is to obtain a new adjacency matrix V via intervention, which constitutes the augmented graph $\mathcal{G}'$, as follows:

$$V = A + \Delta_{A+} - \Delta_{A-} , \qquad (1)$$

where: Δ_{A+} denotes the edges to be added, Δ_{A-} denotes the edges to be deleted. We obtain Δ_{A+} by maximizing the following objective function:

$$
\mathcal{J} = \overbrace{\langle \Theta L, \Delta_{A+} \rangle^2}^{\text{Matching term}} + \overbrace{\epsilon H \left(\Delta_{A+} \right)}^{\text{Entropy regularization}} \\
+ \underbrace{\langle \Theta f, \Delta_{A+} \mathbf{1}_n - a \rangle + \langle \Theta l, \Delta_{A+}^T \mathbf{1}_n - b \rangle}_{\text{Lagrange constraint term}} , \qquad (2)
$$

where Θ is a parameter updated during training, L is the Laplacian matrix of the graph $\mathcal{G}$, ϵ is a weight parameter, f and l are Lagrange multipliers. $\langle P, Q \rangle = \sum_{ij} P_{ij} Q_{ij}$. The vectors a and b are the node degree distributions. $H(\cdot)$ denotes the entropy regularization term, defined as: $H(P) = -\sum_{i,j} P_{i,j} \left(\log \left(P_{i,j} \right) - 1 \right)$. For the calculation of Δ_{A-}, due to space limitations, please refer to (Liu et al. 2022).

Invariance Objective Besides perturbing the non-causal factors S, it is also necessary to keep the causal factors C unchanged. This process can be regarded as an intervention on S, enabling the model to learn invariant representations and thereby improve its generalization ability. Accordingly, the model should satisfy the following equation:

$$P^{do(S=s_i)} (Y \mid C) = P^{do(S=s_j)} (Y \mid C) , \qquad (3)$$

where $do(S = s)$ denotes the intervention on the non-causal factors S. In order to achieve the objective in Equation 3,

based on causal inference theory (Pearl 2009), we can reformulate it as follows:

$$CE \left(C, S = s_i \right) = CE \left(C, S = s_j \right) , \qquad (4)$$

where CE denotes the causal effect between variables. In other words, we need to capture the consistency between the node representations Z^A and Z^V obtained by the encoder f on A and V, respectively. Following existing studies (Mo et al. 2024), we assume that each dimension of the representation follows a Gaussian distribution, and we propose a dimension-level invariance objective, which aims to maintain consistency across views in each dimension. Specifically, by aligning the mean and standard deviation of each dimension of the representations. Formally, the learning objective is defined as:

$$\min_g \sum_i \| Z_i^A - Z_i^V \|_2^2, \, s.t. Std \left(Z_i^A \right) = Std \left(Z_i^V \right) = \lambda , \quad (5)$$

where Z_i^A and Z_i^V are the representations of the embedding matrices in the i-th dimension, and Std is the standard deviation. The first term encourages the means of the embeddings to be aligned in the same dimension, while the second constraint pushes the standard deviations to approach the hyperparameter λ, thereby achieving dimension-level invariance.

Independence Objective To consider a more common scenario where confounding variables exist in the causal graph, leading to spurious correlations between variables, we propose an independence objective to mitigate this challenge. This objective enforces mutual independence among different causal variables, thereby eliminating the correlations between them. In line with existing studies (Mo et al. 2024), we adopt the Hilbert-Schmidt Independence Criterion (HSIC) to measure the independence between variables. When the value of HSIC is 0, it indicates that the two variables are independent. By minimizing the following objective function, we encourage the different dimensions in the representation matrix Z to be mutually independent:

$$\sum_{i \neq j} HSIC \left(Z_i, Z_j \right) = \sum_{i \neq j} \frac{1}{(N-1)^2} \mathrm{Tr} \left(\mathbf{K}_i \mathbf{H} \mathbf{K}_j \mathbf{H} \right) , \quad (6)$$

where $\mathbf{H}$ is the centering matrix $\mathbf{I} - \frac{1}{N} \mathbf{1}\mathbf{1}^\top$, $\mathbf{I}$ is the identity matrix, Z_i represents the i-th dimension of the embedding representation matrix, and $\mathbf{K}_i \in \mathbb{R}^{N \times N}$ and $\mathbf{K}_j \in \mathbb{R}^{N \times N}$ are the kernel matrices of Z_i and Z_j, respectively.

Please note that using complex kernel functions (e.g., Gaussian kernels) in HSIC to measure the independence between different dimensions can result in high spatial complexity, making it difficult to implement in scenarios with large sample sizes and high dimensions. Inspired by previous work (Mialon, Balestriero, and LeCun 2022; Mo et al. 2024), we equivalently treat minimizing the HSIC between different dimensions as minimizing the sum of the off-diagonal elements of the covariance matrix. The proof for this conclusion is as follows:

Let the kernel function be defined as $k \left(Z_{i,a}, Z_{i,b} \right) = \psi \left(Z_{i,a} \right) \psi \left(Z_{i,b} \right)^T$, where ψ is an elementwise mapping
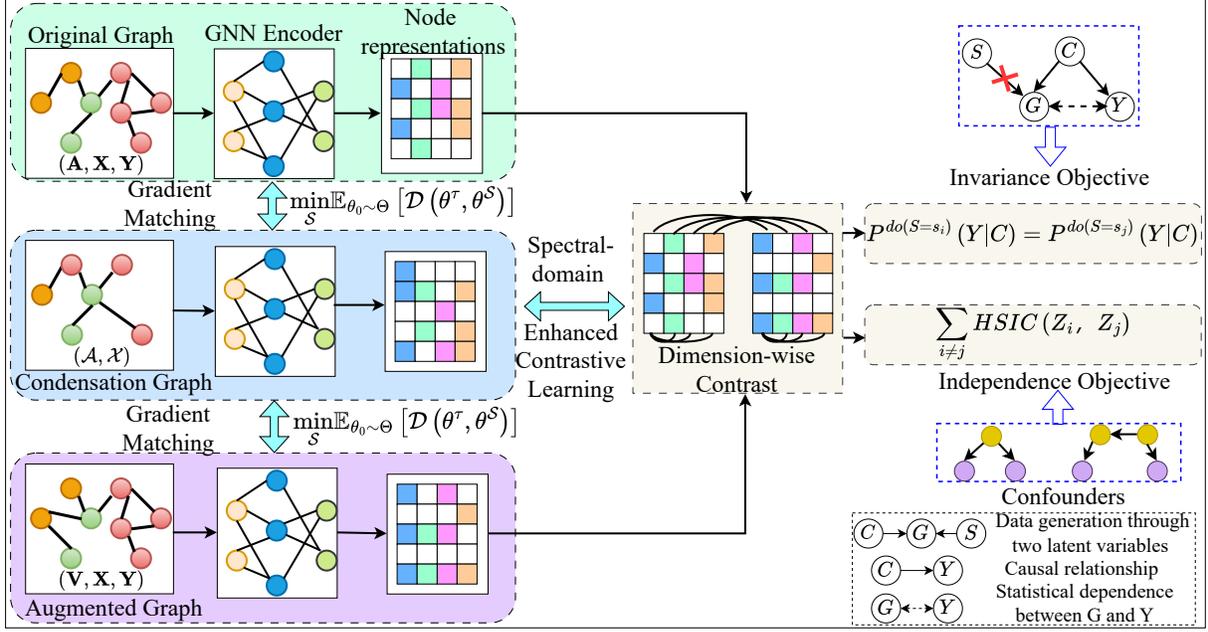
Figure 2: An illustrative diagram of the proposed TGCC framework.

function. We denote the mapping of such projectors on the matrix Z as $Q = \psi(Z) = [\psi(Z_1), \ldots, \psi(Z_d)]$. According to Lemma 1 in (Mialon, Balestriero, and LeCun 2022), we have:

$$
\begin{aligned}
HSIC(Z_i, Z_j) &= \frac{1}{(n-1)^2} \text{Tr}\left(\psi(Z_i)\psi(Z_i)^T \mathbf{H} \psi(Z_j)\psi(Z_j)^T \mathbf{H}\right) \\
&= \frac{1}{(n-1)^2} \left\|\psi(Z_i)^T \mathbf{H} \psi(Z_j)\right\|_F^2 = \|\text{Cov}(\psi(Z_i), \psi(Z_j))\|_F^2 \quad (7) \\
&= \left\|\text{Cov}(Q)_{(i-1)L:iL,(j-1)L:jL}\right\|_F^2,
\end{aligned}
$$

where Tr denotes the trace of a matrix, and Cov represents the covariance between two variables. For complexity considerations and in line with existing studies (Mo et al. 2024), we use a linear kernel, i.e., $g(X) = X$, in this case $Z = Q$, so we obtain:

$$
\sum_{i \neq j} HSIC(Z_i, Z_j) = \sum_{i \neq j} \text{Cov}(Q)_{i,j}^2 = \sum_{i \neq j} \text{Cov}(Z)_{i,j}^2. \quad (8)
$$

We convert the computation of HSIC values between different dimensions into the computation of covariance. By minimizing Eq. 8, the independence between different dimensions can be ensured.

In the this module, we further normalize the embedding matrix along each dimension, and denote the normalized node embeddings as $\bar{\mathbf{Z}}$. Note that since $\|\bar{\mathbf{Z}}\|^2 = 1$, $\min_g \sum_i \|Z_i^A - Z_i^V\|_2^2$ can be equivalently replaced by $\max_g \sum_i \bar{Z}_i^A \cdot \bar{Z}_i^V$, where $\cdot$ denotes the inner product. The s_i represents the standard deviation of the i-th dimension before normalization. Minimizing $\sqrt{\|s_i^A - \lambda\|_2^2}$ encourages the standard deviation to approach the target value λ. The

optimization objective is summarized as follows:

$$
\begin{aligned}
\mathcal{L}_{causal} = &-\alpha \sum_i \tilde{Z}_i^A \cdot \tilde{Z}_i^V + \beta \sum_i \left\{\sqrt{\|s_i^A - \lambda\|_2^2} + \sqrt{\|s_i^V - \lambda\|_2^2}\right\} \\
&+ \gamma \sum_{i \neq j} \left(\text{Cov}(\tilde{Z}^A)_{i,j}^2 + \text{Cov}(\tilde{Z}^V)_{i,j}^2\right),
\end{aligned} \quad (9)
$$

where α, β, and γ are hyperparameters.

Theoretical analysis We have the following theorem to depict the learning process of the TGCC.

Theorem 1. *(Causal Invariance) Given adjacency matrix A and the generated augmentation V, the amplitudes of i-th frequency of A and V are λ_i and γ_i, respectively. With the optimization of $\mathcal{L}_{causal}$, the following upper bound is established:*

$$
\mathcal{L}_{causal} \leq \frac{1}{2} \sum_i \theta_i \|\lambda_i - \gamma_i\|^2 + \sum_j \left[\left(\frac{1}{\sqrt{N}}\|\lambda_j^A\| - \lambda\right)^2 + \left(\frac{1}{\sqrt{N}}\|\gamma_j^Y\| - \lambda\right)^2\right], \quad (10)
$$

where θ_i is an adaptive weight of the i-th term.

Based on Theorem 1, we theoretically prove that TGCC can capture the causal invariance information between A and V. The proof is presented in **Appendix A**.

Graph Contrastive Condensation

We take gradient matching as an example to obtain the condensed graph. It is worth noting that our method is compatible with any condensation approach. We formalize the condensation objective as:

$$
\min_{\mathcal{G}_s} \mathbb{E}_{\theta_0 \sim \Theta}\left[\mathcal{D}\left(\theta^{\mathcal{G}}, \theta^{\mathcal{G}_s}\right)\right], \quad (11)
$$

where θ_0 is the initialization of both $\theta^{\mathcal{G}}$ and $\theta^{\mathcal{G}_s}$, $\mathcal{G}$ is the original graph, and Θ is a specific distribution used for relay model initialization. The expectation over θ_0 aims to improve the robustness of the condensed data $\mathcal{G}_s$ against different parameter initializations (Lei and Tao 2023). $\mathcal{D}(\cdot, \cdot)$ is
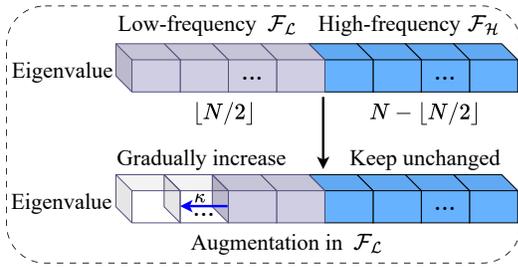
Figure 3: The generation of negative sample.

Datasets	#Classes	#Nodes	#Edges	#Feature
Cora	7	2,708	5,429	1,433
Citeseer	6	3,327	4,732	3,703
Flickr	7	89,250	899,756	500
Ogbn-Arxiv	40	169,343	1,166,243	128
Reddit	210	232,965	57,307,946	602
FinReport (our)	7	4,992	3,568	384

Table 1: Statistics of datasets.

a distance metric. The bi-level optimization objective is approximated by matching the model gradients at each training step. To fully capture both structural and feature information of the graph, we let the training trajectory on the condensed data mimic the training process on both the original and the augmented graphs. Therefore, the optimization objective for graph condensation, denoted as $\mathcal{L}_{\text{cond}}$, is redefined as:

$$
\mathcal{L}_{\text{cond}} = \mathbb{E}_{\theta_0 \backsim \Theta} \left[\begin{array}{l} \sum_{t=1}^{T} \mathcal{D} \left(\nabla_\theta \mathcal{L}^{\mathcal{G}} (\theta_t), \nabla_\theta \mathcal{L}^{\mathcal{G}_s} (\theta_t) \right) \\ + \sum_{t=1}^{T} \mathcal{D} \left(\nabla_\theta \mathcal{L}^{\mathcal{G}'} (\theta_t), \nabla_\theta \mathcal{L}^{\mathcal{G}_s} (\theta_t) \right) \end{array} \right] \quad (12)
$$
$$
\text{s.t. } \theta_{t+1} = \text{opt} \left(\mathcal{L}^{\mathcal{G}_s} (\theta_t) \right),
$$

where T is the number of training steps for the model, $\mathcal{G}' = (X, V, Y)$ is the augmented version of the original graph $\mathcal{G}$, opt $(\cdot)$ is the model parameter optimizer and the parameter of relay model (e.g., GCN) is updated only on $\mathcal{G}_s$.

Spectral-domain Enhanced Contrastive Learning

To inject causally invariant information from the original graph structure into the condensed graph and train a model with stronger generalization ability, we adopt a spectral-domain enhanced contrastive learning strategy. Previous studies (Liu et al. 2022) theoretically proves that GCL can learn invariant information between two augmented graphs, and this invariance is concentrated in the low-frequency components. Additionally, our exploratory experiments also confirm the same conclusion (see **Appendix F**). Therefore, we retain the high-frequency information and perturb the low-frequency components to construct negative samples. The specific construction process is illustrated in Figure 3 and the corresponding expression is as follows:

$$
\hat{\mathcal{L}} = \lambda_{(1-\kappa)*N/2} \mathbf{u}_{(1-\kappa)*N/2} \mathbf{u}_{(1-\kappa)*N/2}^\top + \cdots + \lambda_{\lfloor N/2 \rfloor - 1} \mathbf{u}_{\lfloor N/2 \rfloor - 1} \mathbf{u}_{\lfloor N/2 \rfloor - 1}^\top
$$
$$
+ \lambda_{\lfloor N/2 \rfloor} \mathbf{u}_{\lfloor N/2 \rfloor} \mathbf{u}_{\lfloor N/2 \rfloor}^\top + \cdots + \lambda_N \mathbf{u}_N \mathbf{u}_N^\top,
\quad (13)
$$

where $\mathbf{u}_i$ is the eigenvector corresponding to eigenvalue λ_i, and κ is the proportion of eigenvalues added in descending order. $\hat{\mathcal{L}}$ is the symmetric normalized graph Laplacian, from which the adjacency matrix of negative samples can be further obtained based on the degree matrix.

The core idea of GCL is to train the embedding of a target graph in one augmented view to be as close as possible to the embedding of its positive sample in another aug-

mented view, while being far from those of its negative samples. Models constructed in this way can effectively distinguish between similar and dissimilar graphs. In this work, we adopt the InfoNCE loss as the optimization objective:

$$
\mathcal{L}_{\text{InfoNCE}} = \log \frac{\exp\left(\varphi\left(h^{V_i}, h^{V_j}\right)/t\right)}{\exp\left(\varphi(h^{V_i}, h^{V_j})/t\right) + \sum\limits_{m \neq i} \exp\left(\varphi\left(h^{V_i}, h^{V_m}\right)/t\right)}, \quad (14)
$$

where h^{V_i} and h^{V_j} represent the graph embeddings obtained by applying a readout function to the node feature matrices of the condensed graph and the causally invariant features, respectively. φ is the similarity measure function, such as cosine similarity, and t is the temperature parameter. h^{V_m} denotes the graph embedding corresponding to the constructed negative sample.

Optimization Objective

The overall optimization objective of our proposed method TGCC is summarized as follows:

$$
\mathcal{L} = \mathcal{L}_{causal} + \delta \mathcal{L}_{\text{InfoNCE}} + \eta \mathcal{L}_{\text{cond}}, \quad (15)
$$

where δ and η are hyperparameters that control the importance of each term in the loss function. The complexity analysis of the algorithm can be found in **Appendix D**.

Experiments

Experimental Setup

Datasets We evaluate our method on five public datasets (Cora (Kipf and Welling 2017), Citeseer (Kipf and Welling 2017), Ogbn-arxiv (Hu et al. 2020), Reddit (Hamilton, Ying, and Leskovec 2017), Flickr (Zeng et al. 2020)), and our constructed dataset: **FinReport**. For the node classification task, we follow the settings from GCond (Jin et al. 2022b). For the link prediction task, we follow the public split of the dataset. The statistics of the datasets are provided in Table 1. We report the details of the datasets in **Appendix B**.

Baselines To evaluate the performance of TGCC, we conduct experiments using four types of baselines, including: (1) traditional core-set methods including Random, Herding (Welling 2009), K-Center (Farahani and Hekmatfar 2009), (2) gradient matching methods including DosCond (Jin et al. 2022a), GCond (Jin et al. 2022b) and SGDD (Yang et al. 2023), (3) trajectory matching methods including SFGC (Zheng et al. 2023) and GEOM (Zhang et al. 2024b), and (4) distribution matching methods including GDEM (Liu, Zeng, and Zheng 2024) and CGC (Gao et al. 2025a). The details of evaluated methods are in **Appendix C**.

Datasets	Ratio (r)	Random	Herding	K-Center	DosCond	GCond	SGDD	GDEM	CGC	TGCC	Whole Dataset
Cora	1.3%	58.0±1.4	54.0±0.7	58.0±1.1	66.0±2.4	63.6±1.0	63.7±1.1	67.0±1.2	64.4±1.1	**67.6**±1.8	
	2.6%	55.0±1.4	55.0±1.8	56.0±1.3	64.3±1.3	66.4±0.9	59.7±1.3	67.8±4.5	60.1±0.1	**69.8**±0.4	78.5±1.1
	5.2%	57.0±1.8	56.0±1.9	58.0±1.3	61.5±0.5	61.7±0.7	55.0±0.6	54.4±1.0	60.1±0.1	**65.7**±2.1	
Citeseer	0.9%	52.0±1.3	52.0±1.6	55.0±0.6	60.0±5.3	60.3±1.5	67.1±3.4	72.3±1.8	61.4±2.5	**74.1**±0.2	
	1.8%	52.0±0.5	52.0±1.7	54.0±0.9	50.8±1.5	64.0±2.3	55.1±2.1	72.1±1.0	64.8±0.9	**74.6**±0.9	81.2±1.2
	3.6%	54.0±0.2	53.0±0.8	53.0±1.0	57.5±3.2	68.7±2.6	53.6±0.8	53.5±1.0	59.0±1.2	**69.4**±1.2	
Ogbn-arxiv	0.05%	69.7±1.0	70.3±1.1	70.4±0.7	68.8±0.6	69.2±1.1	68.2±1.2	70.2±0.4	70.2±0.6	**71.0**±0.3	
	0.25%	70.5±0.3	70.6±0.5	70.5±0.3	70.1±0.8	70.7±0.4	68.8±0.8	72.5±0.3	70.7±0.2	70.9±0.7	74.1±0.2
	0.5%	70.8±0.4	70.9±0.5	70.1±0.5	71.5±0.6	71.5±0.3	69.9±0.4	71.4±0.2	71.6±0.0	**72.8**±0.5	
flickr	0.1%	57.2±2.7	54.7±2.9	57.4±1.3	55.3±1.6	70.1±0.4	55.4±1.5	53.3±0.8	54.8±2.6	**70.6**±0.5	
	0.5%	69.6±1.0	68.7±1.5	67.2±0.7	61.6±2.2	70.0±0.4	62.7±1.3	53.3±0.7	52.6±2.1	**71.0**±0.6	74.9±0.1
	1%	70.1±0.4	70.3±0.7	**70.9**±0.6	64.9±1.2	69.7±0.5	66.5±1.1	54.0±1.2	53.4±1.8	68.5±0.7	
reddit	0.05%	59.0±1.7	60.1±1.3	59.6±2.0	55.1±1.3	64.9±2.4	56.1±0.8	61.2±2.8	64.6±1.1	**73.6**±1.1	
	0.1%	61.2±1.3	61.0±2.5	59.6±2.0	57.8±1.6	62.1±1.4	58.4±1.1	62.0±2.6	63.9±2.1	**69.9**±1.0	81.6±0.1
	0.2%	63.4±1.1	65.8±2.1	60.6±1.9	63.5±1.5	63.7±4.5	63.5±1.9	63.9±2.2	62.3±1.7	**72.4**±0.9	
FinReport	2.02%	70.6±1.1	70.7±0.9	71.1±0.8	72.2±1.5	73.8±1.1	70.9±1.6	72.6±1.1	73.1±0.1	**74.6**±0.8	
	4.05%	72.3±0.5	71.9±0.9	71.5±0.3	74.7±0.6	73.8±0.9	72.4±1.7	67.4±3.8	73.2±0.2	**75.4**±0.6	77.0±0.3
	6.07%	72.4±0.4	71.2±0.6	71.5±1.1	74.6±0.7	72.8±0.8	74.5±1.7	72.8±1.9	71.9±0.0	**74.9**±1.1	

Table 2: Link prediction performance (Accuracy% ± std) comparison under the cross-task and single-dataset setting. The best results are highlighted in **bold**, and the runner-up results are **underlined**. $r = m/N$. **Code**: https://github.com/HYJ9999/TGCC.

Implementation Details For each dataset, we perform condensation on 5 graphs using different random seeds and report the average performance. *Single-dataset and single-task scenario*: We train the model using the condensed graph and evaluate it on the original graph. *Cross-dataset and cross-task scenario*: We train the model using the condensed graph, and then train a linear classifier using the downstream data. The parameter settings are provided in **Appendix E**.

Experiments Results

To thoroughly evaluate the performance of TGCC, we conduct experiments in four scenarios: (1) single-task and single-dataset (see **Appendix G1**); (2) cross-dataset; (3) cross-task; and (4) cross-dataset and cross-task.

Performance Comparison on Cross-task Scenario. Following existing studies (Yang et al. 2025), we primarily focus on using node classification tasks for condensation and evaluate link prediction performance on the test set. As shown in Table 2, TGCC achieves either the best or second-best performance across all datasets compared to the baselines. Notably, on the Reddit, our method outperforms the second-best model, GCond, by 13.41%, indicating that the compressed graph effectively preserves causal knowledge beneficial for cross-task scenarios. However, on the Flickr (r= 1%), TGCC performs relatively poorly. A potential reason is the presence of complex confounding relationships and latent variables in the data, which may hinder the accurate extraction of causally invariant features.

Performance Comparison on Cross-dataset Scenario. We compare TGCC with baseline methods under the cross-dataset transfer learning setting. We use Ogbn-arxiv as the source dataset and test the condensed graph on five other target datasets. The results are shown in Table 3. We observe that TGCC achieves the best performance in most cases. These improvements demonstrate the effectiveness of incorporating causal invariant information from the graph structure into universal knowledge extraction, allowing the con-

densed graph to benefit various downstream datasets. Furthermore, TGCC achieves better results on target datasets, such as the FinReport dataset, compared to using the GCN model alone. Therefore, downstream users can achieve similar test performance to expensive GCN models with significantly lower computational costs by training models on the condensed graph and using a simple linear classifier. Additionally, we use FinReport as the source dataset and test the condensed graph on four other target datasets, with results presented in Table 10 in **Appendix G2**, further demonstrating the effectiveness of our method.

Performance Comparison on Cross-task and Cross-dataset Scenario. We compare TGCC with major baseline methods under the cross-dataset and cross-task setting. We use Flickr as the source dataset and evaluate the link prediction performance of the condensed graph on three other target datasets. The results are shown in Table 4. It can be observed that TGCC achieves the best performance. Notably, on the Reddit, our model achieves an AUC improvement of 7.2% and an AP improvement of 7.1%. These improvements further demonstrate the effectiveness of causally invariant information in graph structures, enabling the condensed graph to better support various downstream tasks across different datasets, and also offering a new perspective for the development and training of graph foundation models. More results and analysis can be found in **Appendix G3**.

Generalizability Comparison. To compare the generalizability of different GNN architectures, we evaluated the node classification performance of GC methods under different GNN models, including GCN, SGC, SAGE (Hamilton, Ying, and Leskovec 2017), APPNP (Gasteiger, Bojchevski, and Günnemann 2019), and Cheby (Defferrard, Bresson, and Vandergheynst 2016). We take Reddit as an example to evaluate the performance of different GNN architectures. Table 5 presents the detailed accuracy results. We observe that GCN and SGC outperform other GNNs, as they adopt the same propagation mechanism used in the

Methods	Ogbn-arxiv → Target datasets														
	Cora			Citeseer			Flickr			Reddit			FinReport		
	0.05%	0.25%	0.5%	0.05%	0.25%	0.5%	0.05%	0.25%	0.5%	0.05%	0.25%	0.5%	0.05%	0.25%	0.5%
Random	51.5	58.5	58.0	44.0	56.7	56.3	44.6	<u>44.9</u>	<u>44.9</u>	76.1	84.6	85.5	65.1	<u>66.1</u>	65.9
Herding	48.0	56.6	57.8	48.6	54.6	56.1	44.4	44.7	**45.0**	73.6	83.3	85.2	62.1	66.0	64.1
GCond	57.0	56.7	54.3	55.7	52.1	50.4	44.6	44.6	44.5	85.7	84.5	85.0	66.6	65.9	<u>66.4</u>
SGDD	53.6	<u>58.6</u>	<u>59.0</u>	<u>55.8</u>	<u>57.0</u>	<u>57.5</u>	44.6	44.6	44.6	<u>85.8</u>	<u>85.0</u>	<u>85.9</u>	<u>67.0</u>	65.6	66.3
SFGC	55.3	57.5	<u>59.0</u>	50.0	49.9	55.4	<u>44.9</u>	<u>44.9</u>	44.8	81.4	84.5	**86.2**	65.0	65.7	65.5
GDEM	56.7	58.0	42.9	54.2	46.6	40.4	<u>44.9</u>	44.2	43.1	85.2	82.1	67.3	62.7	62.8	60.9
GEOM	<u>57.2</u>	55.6	58.2	44.4	54.1	56.7	44.6	44.8	<u>44.9</u>	81.5	84.5	<u>85.9</u>	65.8	65.7	66.3
CGC	53.9	58.3	56.7	51.8	53.6	52.0	44.8	44.8	44.5	84.9	84.2	84.8	65.2	66.0	66.3
TGCC	**59.9**	**61.1**	**60.3**	**56.5**	**57.4**	**57.9**	**45.3**	**45.1**	44.8	**86.5**	**85.7**	85.1	**67.9**	**66.3**	**67.2**

Table 3: Node classification performance comparison (Accuracy%) under the cross-dataset scenario.

Methods	Flickr → Target datasets																	
	Cora						Citeseer						Reddit					
	0.1%		0.5%		1.0%		0.1%		0.5%		1.0%		0.1%		0.5%		1.0%	
	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP
Random	51.2	50.6	54.2	52.2	54.7	52.5	54.5	52.4	62.2	56.9	62.8	57.4	55.3	52.9	<u>63.3</u>	<u>57.7</u>	61.3	55.8
GCond	<u>55.6</u>	<u>53.1</u>	<u>57.3</u>	<u>53.1</u>	<u>57.2</u>	53.7	<u>61.4</u>	<u>55.7</u>	<u>63.4</u>	<u>57.8</u>	<u>63.0</u>	<u>57.5</u>	<u>58.4</u>	<u>53.5</u>	62.5	57.2	<u>61.8</u>	<u>56.4</u>
SGDD	51.7	50.9	54.4	52.3	57.0	<u>53.8</u>	53.1	51.6	57.4	54.0	61.6	56.5	50.9	50.4	55.3	52.8	59.1	55.5
DosCond	54.3	52.3	53.3	51.7	55.1	52.7	55.9	53.2	56.0	53.2	61.7	56.7	52.8	51.4	55.7	53.0	60.8	56.1
GDEM	52.7	51.4	51.3	50.7	53.0	51.5	53.8	52.0	52.2	51.1	53.3	51.7	52.4	51.2	50.7	50.3	51.6	50.8
CGC	51.2	50.6	50.6	50.3	50.5	50.3	50.7	50.4	51.2	50.6	51.6	50.8	50.6	50.3	50.0	50.0	50.1	50.1
TGCC	**60.5**	**55.9**	**60.4**	**55.8**	**59.7**	**55.4**	**65.5**	**59.2**	**67.3**	**60.4**	**65.5**	**59.2**	**62.6**	**57.3**	**66.9**	**60.2**	**63.8**	**58.0**

Table 4: Link prediction performance comparison under the cross-task and cross-dataset. AP stands for Average Precision score.

Methods	GCN	SAGE	SGC	APPNP	Cheby	AVG
GCond	87.4	86.2	87.0	84.6	<u>68.3</u>	82.7
SGDD	86.0	83.1	87.5	62.2	59.3	75.6
DosCond	49.5	50.4	53.1	37.6	33.0	44.7
SFGC	78.3	81.1	89.2	**88.0**	58.4	79.0
CGC	<u>88.0</u>	<u>86.9</u>	**90.5**	<u>87.9</u>	61.0	<u>82.9</u>
TGCC	**88.5**	**88.3**	<u>89.5</u>	87.5	**69.4**	**84.6**

Table 5: The generalizability of GC methods on Reddit ($r = 0.05\%$). AVG indicates the average value.

Methods	Flickr			Citeseer		
	$r=0.1\%$	$r=0.5\%$	$r=1.0\%$	$r=0.9\%$	$r=1.8\%$	$r=3.6\%$
w/o CIFE	46.8	46.8	46.9	70.1	69.6	69.5
w/o GCC	45.7	46.8	47.2	70.6	69.3	70.2
w/o ECL	43.5	46.8	47.2	70.3	70.5	68.1
TGCC	**50.2**	**50.2**	**50.3**	**73.0**	**73.3**	**72.8**

Table 6: Node classification performance in ablation study.

Figure 4: The accuracy and condensation time of GC methods on Ogbn-Arxiv ($r = 0.5\%$) and FinReport ($r = 2.02\%$).

feature propagation process during condensation. In addition, TGCC achieves a significant improvement over other baseline methods, indicating that our proposed method can effectively capture causal information in graph data and contribute to better performance on downstream tasks.

Efficiency Comparison. To clearly compare the efficiency of these methods, Figure 4 shows the accuracy of GC methods versus their condensation time on the Ogbn-Arxiv and FinReport. It can be observed that our TGCC achieves the highest test accuracy, and is 3 times and 2 times faster than the SOTA baselines SFGC and GEOM, respectively.
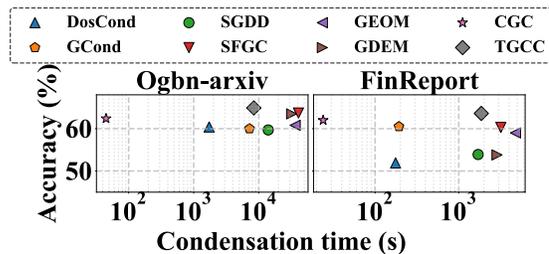
Ablation Study and Sensitivity Analysis. To investigate the impact of different modules in our method, we conduct an ablation study on node classification tasks using Flickr and Citeseer as examples. The variants include: **w/o CIFE** (excluding the causal invariant feature extraction module), **w/o GCC** (excluding the graph condensation module), and **w/o ECL** (excluding the spectral-domain enhanced contrastive Learning). From Table 6, we observe that TGCC achieves the best performance when all components are included. Removing any single module leads to a performance drop, which validates the effectiveness of jointly considering all three components. Due to space limitations, additional ablation studies and detailed parameter sensitivity analyses can be found in **Appendix G4** and **G5**, respectively.

Conclusion

We propose TGCC, a novel causal-invariance-based and transferable graph dataset condensation framework. We first extract causal-invariant knowledge from the graph structure based on causal interventions. Then, through contrastive condensation operations, we extract structural and feature information from the original graph. Finally, we guide the condensation process by using spectral domain enhancement contrastive loss, injecting the causal information. Extensive experiments demonstrate the effectiveness of TGCC.

Acknowledgements

References

Dai, E.; Zhao, T.; Zhu, H.; Xu, J.; Guo, Z.; Liu, H.; Tang, J.; and Wang, S. 2024. A comprehensive survey on trustworthy graph neural networks: Privacy, robustness, fairness, and explainability. *Machine Intelligence Research*, 21(6): 1011–1061.

Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 3844–3852.

Fan, S.; Wang, X.; Mo, Y.; Shi, C.; and Tang, J. 2022. Debiasing graph neural networks via learning disentangled causal substructure. *Advances in Neural Information Processing Systems*, 35: 24934–24946.

Fang, S.; Zhao, K.; Li, G.; and Yu, J. X. 2023. Community search: a meta-learning approach. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, 2358–2371. IEEE.

Farahani, R. Z.; and Hekmatfar, M. 2009. *Facility location: concepts, models, algorithms and case studies*. Springer Science & Business Media.

Gao, X.; Ye, G.; Chen, T.; Zhang, W.; Yu, J.; and Yin, H. 2025a. Rethinking and accelerating graph condensation: A training-free approach with class partition. In *Proceedings of the ACM on Web Conference 2025*, 4359–4373.

Gao, X.; Yu, J.; Chen, T.; Ye, G.; Zhang, W.; and Yin, H. 2025b. Graph condensation: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

Gasteiger, J.; Bojchevski, A.; and Günnemann, S. 2019. Predict then Propagate: Graph Neural Networks meet Personalized PageRank. In *International Conference on Learning Representations*.

Guo, D.; Liu, Z.; and Li, R. 2023. RegraphGAN: A graph generative adversarial network model for dynamic network anomaly detection. *Neural Networks*, 166: 273–285.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, volume 30.

Hu, W.; Fey, M.; Zitnik, M.; Dong, Y.; Ren, H.; Liu, B.; Catasta, M.; and Leskovec, J. 2020. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in neural information processing systems*, volume 33, 22118–22133.

Jin, W.; Liu, X.; Zhao, X.; Ma, Y.; Shah, N.; and Tang, J. 2022a. AUTOMATED SELF-SUPERVISED LEARNING FOR GRAPHS. In *10th International Conference on Learning Representations, ICLR 2022*.

Jin, W.; Zhao, L.; Zhang, S.; Liu, Y.; Tang, J.; and Shah, N. 2022b. Graph Condensation for Graph Neural Networks. In *International Conference on Learning Representations*.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

Lei, S.; and Tao, D. 2023. A comprehensive survey of dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1): 17–32.

Li, J.; Wang, Y.; Zhu, P.; Lin, W.; and Hu, Q. 2024. What matters in graph class incremental learning? An information preservation perspective. *Advances in Neural Information Processing Systems*, 37: 26195–26223.

Li, S.; Wang, X.; Zhang, A.; Wu, Y.; He, X.; and Chua, T.-S. 2022. Let invariant rationale discovery inspire graph contrastive learning. In *International conference on machine learning*, 13052–13065. PMLR.

Liu, J.; Yang, C.; Lu, Z.; Chen, J.; Li, Y.; Zhang, M.; Bai, T.; Fang, Y.; Sun, L.; Yu, P. S.; et al. 2025. Graph foundation models: Concepts, opportunities and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Liu, N.; Wang, X.; Bo, D.; Shi, C.; and Pei, J. 2022. Revisiting graph contrastive learning from the perspective of graph spectrum. *Advances in Neural Information Processing Systems*, 35: 2972–2983.

Liu, Z.; Zeng, C.; and Zheng, G. 2024. Graph data condensation via self-expressive graph structure reconstruction. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1992–2002.

Lu, W.; Guan, Z.; Zhao, W.; Yang, Y.; Sun, Y.; Liang, Z.; Zhan, Y.; and Tao, D. 2025. ProGMLP: A Progressive Framework for GNN-to-MLP Knowledge Distillation with Efficient Trade-offs. *arXiv preprint arXiv:2507.19031*.

Mialon, G.; Balestriero, R.; and LeCun, Y. 2022. Variance covariance regularization enforces pairwise independence in self-supervised representations. *arXiv preprint arXiv:2209.14905*.

Mo, Y.; Wang, X.; Fan, S.; and Shi, C. 2024. Graph contrastive invariant learning from the causal perspective. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 8904–8912.

Pearl, J. 2009. *Causality*. Cambridge university press.

Pearl, J.; Glymour, M.; and Jewell, N. P. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

Sui, Y.; Sun, J.; Wang, S.; Liu, Z.; Cui, Q.; Li, L.; and Wang, X. 2025. A Unified Invariant Learning Framework for Graph Classification. In *KDD*.

Sun, H.; Liu, Z.; Wang, S.; and Wang, H. 2024a. Adaptive attention-based graph representation learning to detect phishing accounts on the ethereum blockchain. *IEEE Transactions on Network Science and Engineering*, 11(3): 2963–2975.

Sun, Q.; Chen, Z.; Yang, B.; Ji, C.; Fu, X.; Zhou, S.; Peng, H.; Li, J.; and Yu, P. S. 2024b. Gc-bench: An open and unified benchmark for graph condensation. *Advances in Neural Information Processing Systems*, 37: 37900–37927.

Wang, Y.; Liu, Z.; Xu, J.; and Yan, W. 2022. Heterogeneous network representation learning approach for ethereum identity identification. *IEEE Transactions on Computational Social Systems*, 10(3): 890–899.

Welling, M. 2009. Herding dynamical weights to learn. In *Proceedings of the 26th annual international conference on machine learning*, 1121–1128.

Xiao, Z.; Wang, Y.; Liu, S.; Hu, B.; Wang, H.; Song, M.; and Zheng, T. 2025. Disentangled condensation for large-scale graphs. In *Proceedings of the ACM on Web Conference 2025*, 4494–4506.

Xu, H.; Zhang, L.; Ma, Y.; Zhou, S.; Zheng, Z.; and Jiajun, B. 2024. A survey on graph condensation. *arXiv preprint arXiv:2402.02000*.

Yang, B.; Sun, Q.; Ji, C.; Fu, X.; and Li, J. 2025. ST-GCond: Self-supervised and Transferable Graph Dataset Condensation. In *The Thirteenth International Conference on Learning Representations*.

Yang, B.; Wang, K.; Sun, Q.; Ji, C.; Fu, X.; Tang, H.; You, Y.; and Li, J. 2023. Does graph distillation see like vision dataset counterpart? *Advances in Neural Information Processing Systems*, 36: 53201–53226.

Yu, X.; Gong, Z.; Zhou, C.; Fang, Y.; and Zhang, H. 2025. Samgpt: Text-free graph foundation model for multi-domain pre-training and cross-domain adaptation. In *Proceedings of the ACM on Web Conference 2025*, 1142–1153.

Zeng, H.; Zhou, H.; Srivastava, A.; Kannan, R.; and Prasanna, V. 2020. GraphSAINT: Graph Sampling Based Inductive Learning Method. In *International Conference on Learning Representations*.

Zhang, S.; Sun, J.; Lin, W.; Xiao, X.; Huang, Y.; and Tang, B. 2024a. Information diffusion meets invitation mechanism. In *Companion Proceedings of the ACM Web Conference 2024*, 383–392.

Zhang, Y.; Zhang, T.; Wang, K.; Guo, Z.; Liang, Y.; Bresson, X.; Jin, W.; and You, Y. 2024b. Navigating Complexity: Toward Lossless Graph Condensation via Expanding Window Matching. In *International Conference on Machine Learning*, 60379–60395. PMLR.

Zhang, Y.; Zhang, Y.; Zhang, Y.; Sang, L.; and Yang, Y. 2025. Unveiling Contrastive Learning's Capability of Neighborhood Aggregation for Collaborative Filtering. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1985–1994.

Zhang, Z.; Wang, X.; Qin, Y.; Chen, H.; Zhang, Z.; Chu, X.; and Zhu, W. 2024c. Disentangled continual graph neural architecture search with invariant modular supernet. In *Forty-first International Conference on Machine Learning*.

Zhao, Y.; Du, H.; Liu, Y.; Wei, S.; Chen, X.; Zhuang, F.; Li, Q.; and Kou, G. 2022. Stock movement prediction based on bi-typed hybrid-relational market knowledge graph via dual attention networks. *IEEE transactions on knowledge and data engineering*, 35(8): 8559–8571.

Zheng, X.; Zhang, M.; Chen, C.; Nguyen, Q. V. H.; Zhu, X.; and Pan, S. 2023. Structure-free graph condensation: From large-scale graphs to condensed graph-free data. *Advances in Neural Information Processing Systems*, 36: 6026–6047.

Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; and He, Q. 2020. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1): 43–76.