Hierarchical Topology Isomorphism Expertise Embedded Graph Contrastive Learning

Jiangmeng Li [1 2*], Yifan Jin[1 3*], Hang Gao[1], Wenwen Qiang[1†], Changwen Zheng[1 3†], Fuchun Sun[1 4]

[1]Science & Technology on Integrated Information System Laboratory, Institute of Software Chinese Academy of Sciences
[2]State Key Laboratory of Intelligent Game
[3]University of Chinese Academy of Sciences
[4]Tsinghua University
{jiangmeng2019,yifan2020,gaohang,qiangwenwen,changwen}@iscas.ac.cn, fcsun@mail.tsinghua.edu.cn

Abstract

Graph contrastive learning (GCL) aims to align the positive features while differentiating the negative features in the latent space by minimizing a pair-wise contrastive loss. As the embodiment of an outstanding discriminative unsupervised graph representation learning approach, GCL achieves impressive successes in various graph benchmarks. However, such an approach falls short of recognizing the topology isomorphism of graphs, resulting in that graphs with relatively homogeneous node features cannot be sufficiently discriminated. By revisiting classic graph topology recognition works, we disclose that the corresponding expertise intuitively complements GCL methods. To this end, we propose a novel hierarchical topology isomorphism expertise embedded graph contrastive learning, which introduces knowledge distillations to empower GCL models to learn the hierarchical topology isomorphism expertise, including the graph-tier and subgraph-tier. On top of this, the proposed method holds the feature of plug-and-play, and we empirically demonstrate that the proposed method is universal to multiple state-of-the-art GCL models. The solid theoretical analyses are further provided to prove that compared with conventional GCL methods, our method acquires the tighter upper bound of Bayes classification error. We conduct extensive experiments on real-world benchmarks to exhibit the performance superiority of our method over candidate GCL methods, e.g., for the real-world graph representation learning experiments, the proposed method beats the state-of-the-art method by 0.23% on unsupervised representation learning setting, 0.43% on transfer learning setting. Our code is available at https://github.com/jyf123/HTML.

1 Introduction

Unsupervised graph representation learning aims to capture the discriminative information from graphs without manual annotations. The intuition behind such a learning approach is that the covariate shift problem (Heckman 1979; Shimodaira 2000) occurs when the data distributions in the source labeled and target unlabeled domains are different. Thus, the *identically distributed* assumption is violated, halting the forthright application of the model trained in the supervised manner on the target unlabeled dataset. In view of that, the contrastive learning paradigm has achieved impressive successes in a spectrum of practical fields, such as computer vision (Wu et al. 2018; Chen et al. 2020; Qiang et al. 2022), natural language processing (Gao, Yao, and Chen 2021; Qu et al. 2021), signal processing (Yao et al. 2022; Tang et al. 2022), etc. As tractable surrogates, the contrastive approaches further establish promising capacity in the field of unsupervised graph learning, namely GCL methods (You et al. 2020). The primary principle of GCL is jointly differentiating features of heterogeneous samples while gathering features of homogeneous samples.

A major challenge for state-of-the-art (SOTA) GCL methods is sufficiently exploring *heterogeneous* discriminative information. Orthogonal to the natural data structures, e.g., images and videos, as the artificially derived discrete data structure, graphs contain two critical ingredients of potential discriminative information: feature-level information and topology-level information. Benefiting from the powerful non-linear mapping capability of graph neural networks (GNNs) (Kipf and Welling 2016; Velickovic et al. 2017; Gao et al. 2023), GCL methods can practically explore feature-level discriminative information. However, recent works (Xu et al. 2018; Wijesinghe and Wang 2022) demonstrate that compared with topology-expertise-based approaches, GNN-based GCL methods can only model relatively limited topology-level discriminative information, e.g., (Xu et al. 2018) empirically substantiate that specific GNN-based methods (Kipf and Welling 2016; Hamilton, Ying, and Leskovec 2017) fall short of discriminating the topology structures of graphs that the Weisfeiler-Lehman-based approach can explicitly discriminate. To this end, we are primarily dedicated to exploring the difference in knowledge between the topology-expertise-based approaches and GNN-based GCL methods with respect to capturing the discriminative information. As demonstrated in Figure 1, the experimental exploration attests that canonical GNN-based GCL methods *cannot* effectively learn the topology expertise, and accordingly, introducing such graph expertise into the GNN-based GCL method incurs a precipitous rise in the discriminative performance. Concretely, we empirically derive a conclusion: the topology expertise is complementary to GNN-based GCL methods, and prudently aggregating the

[*]These authors contributed equally.
[†]Corresponding author.

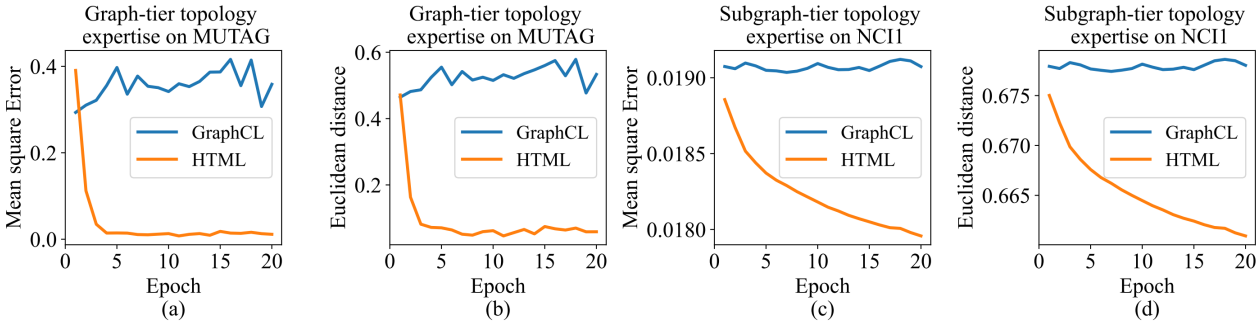

Figure 1: Motivating experiments to explore whether canonical GCL methods can learn the topology expertise on graph benchmarks. For the fair comparison principle, we adopt two metrics to measure the error or similarity between the candidate methods and the topology expertise, containing the graph-tier and subgraph-tier, during training. The volatile result curves show that canonical GCL methods, e.g., GraphCL, can barely learn topology expertise. Yet, the consistent result curves in the downtrend substantiate that the proposed method, i.e., HTML, can indeed introduce the topology expertise in GCL methods.

topology-level and feature-level discriminative information can explicitly improve the model's performance on unsupervised graph prediction.

Considering the aforementioned conclusion, we explore implicitly introducing the graph topology expertise into GCL models and thus propose a plug-and-play method, namely *hierarchical topology isomorphism expertise embedded graph contrastive learning* (HTML), which leverages the knowledge distillation technique to prompt GCL models to learn the topology expertise. Specifically, by revisiting the graph topology expertise, we disclose that the principal objective of such expertise is determining the *isomorphism* of candidate graph units, and the determination of whether candidate graphs are graph-tier isomorphic is identified as a *non-deterministic polynomial immediate problem* (NPI problem) (Babai 2015). Yet, state-of-the-art graph isomorphism studies demonstrate that there exist certain closed-form analytical solutions in polynomial time, e.g., Weisfeiler-Lehman (WL) test (Weisfeiler and Leman 1968), which can fit most cases in the graph isomorphism problem, such that introducing the guidance of graph topology isomorphism expertise into GCL methods is theoretically feasible. Considering there are still certain cases that cannot be solved by the deterministic methods in polynomial-time, we propose to introduce the graph isomorphism expertise in an implicit instead of explicit manner. Accordingly, the topology isomorphism expertise adheres to the hierarchical principle and can be further divided into the graph-tier and the subgraph-tier. We elaborate that the graph-tier topology isomorphism expertise dedicates to globally classifying graphs with different topological structures, while the subgraph-tier topology isomorphism expertise focuses on discriminating graphs regarding certain critical subgraphs, e.g., functional groups for a chemical molecule, by leveraging the weighting strategy. Concretely, in view of the conducted exploration, we reckon that HTML can generally improve GCL methods with confidence. The thoroughly proved theoretical analyses are provided to demonstrate that HTML can tighten the upper bound of Bayes classification error acquired by conventional GCL methods. The extensive experiments on benchmarks demonstrate that the proposed method is simple yet effective. In detail, HTML generally exhibits performance superiority over candidate GCL methods under the unsupervised learning experimental setting. As a plug-and-play module, HTML consistently promotes the performance of baselines by significant margins. **Contributions**:

- We disclose the differences between the topology isomorphism expertise approach and the GNN-based GCL method and further introduce an empirical exploration, which demonstrates that prudently aggregating the topology isomorphism expertise and the knowledge contained by canonical GCL methods can effectively improve the model performance.
- We propose the novel HTML, orthogonal to existing methods, to introduce the hierarchical topology isomorphism expertise learning into the GCL paradigm, thereby promoting the model to capture discriminative information from graphs in a plug-and-play manner.
- The performance rises of GCL methods incurred by leveraging the proposed HTML are theoretically supported since HTML can tighten the Bayes error upper bound of canonical GCL methods on graph learning benchmarks.
- Empirically, experimental evaluations on various real-world datasets demonstrate that HTML yields wide performance boosts on conventional GCL methods.

2 Related Works

Graph neural networks (GNNs) are a group of neural networks which can effectively encode graph-structured data. Since the inception of the first GNNs model, multiple variants have been proposed, including GCN (Kipf and Welling 2016), GAT (Velickovic et al. 2017), GraphSAGE (Hamilton, Ying, and Leskovec 2017), GIN (Xu et al. 2018), and others. These models learn discriminative graph representations guided by the labels of graph data. Since the annotation of graph data, such as the categories of biochemical molecules, requires the support of expert knowledge, it is

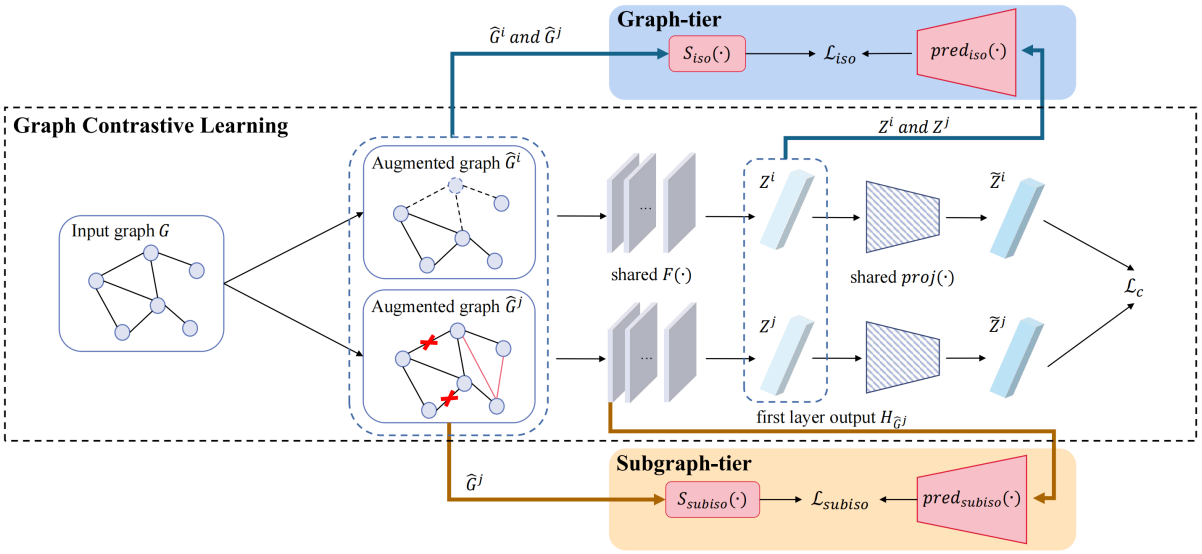

Figure 2: Architecture of the proposed HTML based on a canonical GCL approach.

not easy to obtain large-scale annotated graph datasets (You et al. 2020). This leads to the limitation of supervised graph representation learning.

As one of the most effective self-supervised methods, contrastive learning (CL) aims to learn discriminative representations from unlabelled data (Li et al. 2022a). With the principle of closing positive and moving away from negative pairs, CL methods, such as SimCLR (Chen et al. 2020), MoCo (He et al. 2020), BYOL (Grill et al. 2020), MetAug (Li et al. 2022b), and Barlow Twins (Zbontar et al. 2021), have achieved outstanding success in the field of computer vision (Wu et al. 2018; Chen et al. 2020; Qiang et al. 2022).

Benefiting from GNNs and CL methods, GCL is proposed (Velickovic et al. 2019; Sun et al. 2020; Hassani and Ahmadi 2020; You et al. 2020). Due to the difference between graph and image, the GCL methods focus more on constructing graph augmentation. Specifically, GraphCL (You et al. 2020) first introduces the perturbation invariance for GCL and proposes four types of graph augmentation: node dropping, edge perturbation, attribute masking, and subgraph. Noting that using a complete graph leads to high time and space utilization, Subg-Con (Jiao et al. 2020) proposes to sample subgraphs to capture structural information. To enrich the semantic information in the sampled subgraphs, MICRO-Graph (Zhang et al. 2020) proposes to generate informative subgraphs by learning graph motifs. Besides, the choice of graph augmentation requires rules of thumb or trial-and-error, resulting in time-consuming and labor-intensive. Thus, JOAO (You et al. 2021) introduces a bi-level optimization framework that automatically selects data augmentations for particular graph data. RGCL (Li et al. 2022c) argues that randomly destroying the properties of graphs will result in the loss of important semantic information in the augmented graph and proposes a rationale-aware graph augmentation method. For graph invariance in graph augmentation, SPAN (Lin, Chen, and Wang 2023) finds that previous studies have only performed topology augmentation in the spatial domain, therefore introducing a spectral perspective to guide topology augmentation. Noting that existing GCL algorithms ignore the intrinsic hierarchical structure of the graph, HGCL (Ju et al. 2023) proposes Hierarchical GCL consisting of three components: node-level CL, graph-level CL, and mutual CL. Despite its attention to the neglect of the intrinsic hierarchical structure, compared to HTML, it still focuses only on the semantic information, i.e., the feature-level information. Another important issue with GCL is negative sampling. BGRL (Thakoor et al. 2022) proposes a method that uses simple graph augmentation and does not have to construct negative samples while effectively scaling to large-scale graphs. To address the issue of sampling bias in GCL, PGCL (Lin et al. 2021) proposes a negative sampling method based on semantic clustering. In contrast to previous approaches, HTML introduces topology-level information into GCL, which has never been approached from a novel perspective, i.e., neither constructing graph augmentation nor improving negative sampling.

3 Preliminary

GCL aims at maximizing the mutual information of positive pairs while minimizing that of negative pairs, which is capable of learning discriminative graph representations without annotations. Generally, GCL methods consist of three components, as shown in the middle of Figure 2.

Graph augmentation. Let $G = (V, E)$ denotes a graph, V is the node set, and E is the edge set. Graph G is first augmented into two different views $\hat{G}^i$ and $\hat{G}^j$ by node dropping, edge perturbation (You et al. 2020), rationale sampling (Li et al. 2022c), etc.

Graph encoder. The graph encoder, e.g., specific GNNs with shared parameters, is used to extract graph representations Z^i and Z^j for views $\hat{G}^i$ and $\hat{G}^j$.

Contrastive loss function. Before computing the con-

trastive loss, a projection head is commonly applied to map Z^i and Z^j into $\tilde{Z}^i$ and $\tilde{Z}^j$. Then, by considering two different views of the same graph, such as $\hat{G}^i$ and $\hat{G}^j$, as positive pairs while views of different graphs are negative pairs and adopting the normalized temperature-scaled cross-entropy loss (NT-Xent) (van den Oord, Li, and Vinyals 2018; Wu et al. 2018; Sohn 2016), the loss can be computed as follows:

$$\mathcal{L}_c = \frac{1}{N}\sum_{n=1}^{N} log \frac{-\exp\left(\text{sim}\left(\tilde{Z}_{n,i}, \tilde{Z}_{n,j}\right)/\tau\right)}{\sum_{n'=1,n'\neq n}^{N}\exp\left(\text{sim}\left(\tilde{Z}_{n,i}, \tilde{Z}_{n',j}\right)/\tau\right)}, \tag{1}$$

where $\text{sim}(\cdot,\cdot)$ denotes cosine similarity (Xia, Zhang, and Li 2015), which can be calculated as $\text{sim}(\tilde{Z}_{n,i}, \tilde{Z}_{n,j}) = \tilde{Z}_{n,i}\tilde{Z}_{n,j}^{\top} / \left\|\tilde{Z}_{n,i}\right\| \left\|\tilde{Z}_{n,j}^{\top}\right\|$, N is the number of graphs, and τ is the temperature coefficient.

4 Methodology

To enable GNN-based GCL methods to model topology-level discriminative information, we implicitly incorporate topology isomorphism expertise into GCL models at both the graph-tier and the subgraph-tier. The overall framework is illustrated in Figure 2.

4.1 Learning Graph-tier Topology Isomorphism Expertise

With the aim of empowering the graph encoder $F(\cdot)$ to learn graph-tier topology isomorphic expertise, we propose an expert system $\mathcal{S}_{iso}(\cdot)$ to measure the isomorphic similarity between two graphs. The graph-tier topology isomorphism measures the topological equivalence of two graphs. Determining whether two graphs are graph-tier topology isomorphic remains an NPI problem. Among many methods (Corneil and Gotlieb 1970; McKay and Piperno 2014; Babai 2015; Weisfeiler and Leman 1968) for this problem, the WL test (Weisfeiler and Leman 1968), especially the 1-dimensional WL test, is a relatively effective method that can correctly deterministically confirm whether two graphs are graph-tier isomorphic in most cases (Babai and Kucera 1979). Specifically, given a graph with pre-defined node labels, the 1-dimensional WL test iteratively 1) aggregates the labels of neighbors to the original node labels, 2) updates node labels by the label compression and relabeling, and eventually obtains the set of node labels after several iterations. Accordingly, we build the proposed learning procedure of graph-tier topology isomorphism expertise. Given two augmented graphs $\hat{G}^i$ and $\hat{G}^j$ of graph G sampled *i.i.d* from a dataset with N graphs, the expert system $\mathcal{S}_{iso}(\cdot)$ first performs the 1-dimensional WL test to obtain the node label sets A^i and A^j, respectively. In practice, the number of iterations of the 1-dimensional WL test is the same as the number of layers of the encoder $F(\cdot)$. The graph-tier isomorphic similarity between the graphs can be computed by adopting the Jaccard Coefficient (Jaccard 1912):

$$y_{iso} = \frac{\left|A^i \cap A^j\right|}{\left|A^i \cup A^j\right|}, \tag{2}$$

where y_{iso} denotes the graph-tier topology isomorphic similarity between $\hat{G}^i$ and $\hat{G}^j$.

To introduce the graph-tier topology isomorphism expertise to GCL methods, we propose a self-supervised distillation task. Specifically, we train $F(\cdot)$ to predict the isomorphic similarity of $\hat{G}^i$ and $\hat{G}^j$ based on the graph representation obtained by $F(\cdot)$. For the prediction, the representations of $\hat{G}^i$ and $\hat{G}^j$ obtained by $F(\cdot)$ are concatenated and fed into a multi-layer perceptron (MLP), which can be treated as a prediction head. This process can be formulated by

$$\hat{y}_{iso} = \text{MLP}\left([Z^i; Z^j]\right). \tag{3}$$

We adopt the mean square error (Allen 1971) to measure the ability of $F(\cdot)$ to learn the topology isomorphism expertise in the graph-tier and promote $F(\cdot)$ to capture the topology discriminative information:

$$\mathcal{L}_{iso} = \frac{1}{N}\sum_{n=1}^{N}\left(y_{iso}^n - \hat{y}_{iso}^n\right)^2. \tag{4}$$

4.2 Learning Subgraph-tier Topology Isomorphism Expertise

Graph-tier topology isomorphism expertise treats one graph as a whole, while in practice, specific subgraphs of a graph provide relatively significant contributions in the discrimination of graphs on certain downstream tasks. For instance, the functional groups of a chemical molecule have a decisive influence on certain properties. Therefore, we argue that learning the subgraph-tier topology isomorphism expertise can further promote the discriminative performance of GCL methods. Inspired by (Wijesinghe and Wang 2022), we introduce an expert system $\mathcal{S}_{subiso}(\cdot)$ to guide $F(\cdot)$ to explore the subgraph-tier topology isomorphism expertise. To capture the fine-grained topology information in a graph, *structural coefficients* (Wijesinghe and Wang 2022) are introduced. By modeling the structural characteristics of the *overlap subgraph* between adjacent nodes, the structural coefficients can satisfy the properties of local closeness, local denseness, and isomorphic invariance. Specifically, local closeness presents that the structural coefficients are positively correlated with the number of the overlap subgraph nodes; local denseness presents the positive correlation between the structural coefficients and the number of the overlap subgraph edges; isomorphic invariant presents that the structural coefficients are identical when the overlap subgraphs are isomorphic. Given a node v in a graph G sampled *i.i.d* from a dataset with N graphs, the *neighborhood subgraph* of node v, denoted as O_v, is composed by itself, the nodes directly connected to v, and the corresponding edges. For two adjacent nodes v and u, the overlap subgraph between them is defined as the intersection of their neighborhood subgraphs, i.e., $O_{vu} = O_v \cap O_u$. Formally, the structural coefficient w_{vu} of v and u can be implemented by

$$w_{vu} = \frac{|N^{E_{vu}}|}{|N^{V_{vu}}| \cdot |N^{V_{vu}} - 1|}|N^{V_{vu}}|^{\lambda}, \tag{5}$$

where $N^{V_{vu}}$ and $N^{E_{vu}}$ denote the aggregated numbers of nodes and edges of O_{vu}, respectively, and $\lambda{>}0$ is a hyperparameter. Then, w_{vu} is normalized by $\tilde{w}_{vu} = \frac{w_{vu}}{\sum_{u\in\mathcal{N}(v)} w_{vu}}$,

where $\mathcal{N}(v)$ is the neighbor set of v. For guiding the encoder $F(\cdot)$ to learn the fine-grained topology expertise, the structural coefficient matrix of G is derived to constitute the subgraph-tier topology isomorphism expertise y_{subiso}.

We further leverage the node representation H_G obtained from the first layer of $F(\cdot)$ to fit the structural coefficient matrix:

$$\hat{y}_{subiso} = \text{MLP}\left(\text{AUTOCOR}\left(\text{MLP}\left(H_G\right)\right)\right), \tag{6}$$

where $\hat{y}_{subiso}$ denotes the predicted subgraph-tier topology isomorphism, MLPs can be synthetically treated as the projection head, and AUTOCOR $(\cdot)$ denotes the autocorrelation matrix computation function (Berne, Boon, and Rice 1966).

The mean square error loss is used to align the predictions of the expert system and $F(\cdot)$, thereby deriving

$$\mathcal{L}_{subiso} = \frac{1}{N}\sum_{n=1}^{N}\left(y^n_{subiso} - \hat{y}^n_{subiso}\right)^2. \tag{7}$$

4.3 Training Pipeline

To jointly learn the graph-tier and subgraph-tier topology isomorphism expertise, we introduce $\mathcal{L}_{iso}$ and $\mathcal{L}_{subiso}$ into the GCL paradigm and derive the ultimate objective as follows:

$$\mathcal{L} = \mathcal{L}_c + \alpha\mathcal{L}_{iso} + \beta\mathcal{L}_{subiso}, \tag{8}$$

where α and β are coefficients balancing the impacts of $\mathcal{L}_{iso}$ and $\mathcal{L}_{subiso}$ during training, respectively. The detailed training process is shown in Algorithm 1.

Algorithm 1: HTML

Input: two augmented graphs $\hat{G}^i$ and $\hat{G}^j$ of graph G, graph encoder $F(\cdot)$, projection head $proj(\cdot)$, node representation $H_{\hat{G}^j}$ obtained by the first layer of $F(\cdot)$, temperature parameter τ, and hyperparameters α, β.

for t-th training iteration **do**

$Z^i = F(\hat{G}^i), Z^j = F(\hat{G}^j)$

$\tilde{Z}^i = proj(Z^i), \tilde{Z}^j = proj(Z^j)$

$\#\ graph\ contrastive\ learning\ loss$

$\mathcal{L}_c = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{\exp\left(\text{sim}\left(\tilde{Z}_{n,i},\tilde{Z}_{n,j}\right)/\tau\right)}{\sum_{n'=1,n'\neq n}^{N}\exp\left(\text{sim}\left(\tilde{Z}_{n,i},\tilde{Z}_{n',j}\right)/\tau\right)}$

$\#\ learning\ graph-tier$

$\#\ topology\ isomorphism\ expertise$

Execute expert system $\mathcal{S}_{iso}(\cdot)$ to obtain graph-tier topology isomorphism expertise y_{iso} of $\hat{G}^i$ and $\hat{G}^j$.

$\hat{y}_{iso} = \text{MLP}\left(\left[Z^i; Z^j\right]\right)$

$\mathcal{L}_{iso} = \frac{1}{N}\sum_{n=1}^{N}\left(y^n_{iso} - \hat{y}^n_{iso}\right)^2$

$\#\ learning\ subgraph-tier$

$\#\ topology\ isomorphism\ expertise$

Execute expert system $\mathcal{S}_{subiso}(\cdot)$ to obtain subgraph-tier topology isomorphism expertise y_{subiso} of $\hat{G}^i$.

$\hat{y}_{subiso} = \text{MLP}\left(\text{AUTOCOR}\left(\text{MLP}\left(H_{\hat{G}^i}\right)\right)\right)$

$\mathcal{L}_{subiso} = \frac{1}{N}\sum_{n=1}^{N}\left(y^n_{subiso} - \hat{y}^n_{subiso}\right)^2$

$\#\ training\ loss$

$\mathcal{L} = \mathcal{L}_c + \alpha\mathcal{L}_{iso} + \beta\mathcal{L}_{subiso}$

Update encoder $F(\cdot)$ via $\mathcal{L}$.

end for

5 Theoretical Analyses

Within the canonical self-supervised GCL paradigm, we demonstrate that compared with conventional graph contrastive methods, the proposed HTML achieves a relatively tighter Bayes error bound, thereby validating the performance superiority of HTML from the theoretical perspective. For conceptual simplicity, we demonstrate the Bayes error bound on a binary classification problem within a 1-dimensional latent space. First, we hold an assumption:

Assumption 5.1. *Considering the universality of Gaussian distributions in statistics, we assume that the prior distributions of candidate categories, i.e., $\mathcal{P}_{\mathcal{C}_1}$ and $\mathcal{P}_{\mathcal{C}_2}$, adhere to the central limit theorem (Rosenblatt 1956) on the target bi-classification task $\mathcal{T}^b$, such that $\{X_i\}_{i=0}^{N_{\mathcal{C}_1}} \sim \mathcal{P}_{\mathcal{C}_1} = \mathbf{N}\left(\mu_{\mathcal{C}_1}, \sigma_{\mathcal{C}_1}\right)$ and $\{X_i\}_{i=0}^{N_{\mathcal{C}_2}} \sim \mathcal{P}_{\mathcal{C}_2} = \mathbf{N}\left(\mu_{\mathcal{C}_2}, \sigma_{\mathcal{C}_2}\right)$. This assumption holds on the condition that the data variables are composed of a wealth of statistically independent random factors on graph benchmarks.*

Then, as the thorough distributions of candidate categories $\mathcal{C}_1$ and $\mathcal{C}_2$ are agnostic, we demonstrate the probability of error for task $\mathcal{T}^b$ as follows:

Lemma 5.2. *Given the unknown mean vectors $\mu_{\mathcal{C}_1}$ and $\mu_{\mathcal{C}_2}$ and $N_S = N_{\mathcal{C}_1} + N_{\mathcal{C}_2}$ labeled training samples, we can recap the optimal Bayes decision rule to construct the decision boundary based on a plain 0-1 loss function and derive the probability of error as*

$$\mathcal{R}_{\mathcal{T}^b}\left(N_S, D\right) = \int_{\kappa(D)}^{\infty}\frac{1}{\sqrt{2\pi}}\cdot e^{-\frac{\rho^2}{2}}d\rho, \tag{9}$$

where D denotes the dimensionality of graph representations learned by a neural network and

$$\kappa\left(D\right) = \frac{\sum_{\nu=1}^{D}\frac{1}{\nu}}{\sqrt{\frac{D}{N_S} + \left(1 + \frac{1}{N_S}\right)\cdot\sum_{\nu=1}^{D}\frac{1}{\nu}}}. \tag{10}$$

Lemma 5.2 states that the probability of error is directly related to the training sample size N_S and the representation dimensionality D. According to the experimental setting of graph benchmarks, N_S is fixed among comparisons. The proposed HTML is a plug-and-play method that fits various GCL methods without mandating a shift in the dimensionality of the learned representations, thereby preserving a constant D. Therefore, we attest that the theoretical analyses on the Bayes errors of the canonical GCL method and HTML can be conducted in a fair manner. Considering the complexity gap between the cross-entropy loss and the 0-1 loss, we introduce an approach for estimating the bounds of Bayes error based on divergence (Jain, Duin, and Mao 2000) by following the principle of Mahalanobis distance measure (De Maesschalck, Jouan-Rimbaud, and Massart 2000) as follows:

Lemma 5.3. *Given the mean vectors $\mu_{\mathcal{C}_1}$, $\mu_{\mathcal{C}_2}$ and the covariance matrices $\sigma_{\mathcal{C}_1}$, $\sigma_{\mathcal{C}_2}$ for the hypothesis Gaussian distributions $\mathcal{P}_{\mathcal{C}_1}$ and $\mathcal{P}_{\mathcal{C}_2}$, respectively, we denote $\bar{\sigma}$ be the non-singular average covariance matrix over $\sigma_{\mathcal{C}_1}$ and $\sigma_{\mathcal{C}_2}$, i.e., $\bar{\sigma} = P(\mathcal{C}_1)\cdot\sigma_{\mathcal{C}_1} + P(\mathcal{C}_2)\cdot\sigma_{\mathcal{C}_2}$, where $P(\mathcal{C}_1)$ and $P(\mathcal{C}_2)$*

are the probability densities corresponding to the prior distributions $\mathcal{P}_{\mathcal{C}_1}$ and $\mathcal{P}_{\mathcal{C}_2}$. We can conduct the divergence estimation on the upper bound of Bayes error $\mathcal{R}_{\mathcal{T}^b}$ by

$$\mathcal{R}_{\mathcal{T}^b} \leq \frac{2P(\mathcal{C}_1)P(\mathcal{C}_2)}{1 + P(\mathcal{C}_1)P(\mathcal{C}_2) \cdot [(\mu_{\mathcal{C}_1} - \mu_{\mathcal{C}_2})^\top \bar{\sigma}^{-1}(\mu_{\mathcal{C}_1} - \mu_{\mathcal{C}_2})]}, \tag{11}$$

Holding Lemma 5.3, we can deduce the upper bound of the error gap between the canonical GCL method and HTML as follows:

Proposition 5.4. *(Tumer 1996) states that the Bayes optimum decision is the loci of all samples $X^\star$ satisfying $P(\mathcal{C}_1|X^\star) = P(\mathcal{C}_2|X^\star)$. Then, from the perspective of classification, we can derive that the predicted decision boundary of the canonical GCL model f is the loci of the available samples X^A satisfying $f(\mathcal{C}_1|X^A) = f(\mathcal{C}_2|X^A)$, and there exists a constant gap X^Δ, i.e., $\exists X^\Delta = X^A - X^\star$.*

Considering Proposition 5.4, we deduce that

Corollary 5.5. *Given an independent and sole GCL model f for classification and the available training samples X^A, we can get $f(\mathcal{C}|X^A) = P(\mathcal{C}|X^A) + \mathcal{E}(\mathcal{C}, f, X^A)$, where $\mathcal{E}(\cdot,\cdot,\cdot)$ is the inherent error. Following (Tumer 1996), $\mathcal{E}(\mathcal{C}, f, X^A)$ is implemented by*

$$\mathcal{E}(\mathcal{C}, f, X^A) = \Phi_{\mathcal{C}} + \Psi_{\mathcal{C},f,X^A}, \tag{12}$$

where $\Phi_{\mathcal{C}}$ is constant for the category $\mathcal{C}$, while $\Psi_{\mathcal{C},f,X^A}$ is additionally associated with the model f and the training samples X^A.

Considering Equation 12, we can arbitrarily discard $\Phi_{\mathcal{C}}$ during the deduction of the error gap between the canonical GCL model and HTML since the available training samples and categories are constant on graph benchmarks, such that $\Phi_{\mathcal{C}}$ is constant, and only $\Psi_{\mathcal{C},f,X^A}$ varies along with the change of model.

The proposed HTML can be treated as an implicit multi-loss ensemble learning model (Rajaraman, Zamzmi, and Antani 2021), and the features learned by back-propagating detached losses are contained by multiple-dimensional subsets of the ultimate representations. The classification weights, following the linear probing scheme, perform the implicitly weighted ensemble for representations. Therefore, HTML can be decomposed into the canonical GCL model f and the plug-and-play hierarchical topology isomorphism expertise learning model $f_\star$, such that we derive the error gap between the candidate models as follows:

Theorem 5.6. *Given the decomposed HTML models f and $f_\star$ for classification, the Bayes error gap between the proposed HTML and the canonical GCL model can be approximated by*

$$\mathcal{R}_{\mathcal{T}^b}(f) - \mathcal{R}_{\mathcal{T}^b}(\{f, f_\star\}) = \frac{3\delta^2_{\Psi_{\mathcal{C}_1,f,X^A}} + 3\delta^2_{\Psi_{\mathcal{C}_2,f,X^A}} - \delta^2_{\Psi_{\mathcal{C}_1,f_\star,X^A}} - \delta^2_{\Psi_{\mathcal{C}_2,f_\star,X^A}}}{8|\nabla_{X^A=X^\star} P(\mathcal{C}_1|X^A) - \nabla_{X^A=X^\star} P(\mathcal{C}_2|X^A)|}. \tag{13}$$

According to the experimental setting of graph benchmarks, the inherent error Ψ is associated and only associated with models f and $f_\star$, and both $\mathcal{R}_{\mathcal{T}^b}(f)$ and $\mathcal{R}_{\mathcal{T}^b}(\{f, f_\star\})$ are associated with Ψ, such that the Bayes error gap $\mathcal{R}_{\mathcal{T}^b}(f) - \mathcal{R}_{\mathcal{T}^b}(\{f, f_\star\})$ is solely dependent to the candidate models f and $f_\star$, which further substantiates that the theoretical analyses are imposed in an impartial scenario. Holding the empirically solid hypothesis that the differences of inherent errors, i.e., $\left\langle \Psi_{\mathcal{C}_1,f,X^A}, \Psi_{\mathcal{C}_1,f_\star,X^A} \right\rangle$ and $\left\langle \Psi_{\mathcal{C}_2,f,X^A}, \Psi_{\mathcal{C}_2,f_\star,X^A} \right\rangle$, are bounded, that is, the performance gap of the GCL model and HTML is not dramatically large, we can derive the following:

Corollary 5.7. *Given the candidate models f and $f_\star$, the available training samples X^A, and the labels $\mathcal{C}$ of target downstream task $\mathcal{T}^b$, we suppose that the corresponding inherent errors are bounded, i.e., $-\epsilon_f \leq \Psi_{\mathcal{C},f,X^A} \leq \epsilon_f$ and $-\epsilon_{f_\star} \leq \Psi_{\mathcal{C},f_\star,X^A} \leq \epsilon_{f_\star}$. If $\epsilon_{f_\star} \leq \sqrt{3} \cdot \epsilon_f$ holds, the positive definiteness of the Bayes error gap between $\mathcal{R}_{\mathcal{T}^b}(f)$ and $\mathcal{R}_{\mathcal{T}^b}(\{f, f_\star\})$ can be satisfied, i.e., we can establish that $\mathcal{R}_{\mathcal{T}^b}(f) - \mathcal{R}_{\mathcal{T}^b}(\{f, f_\star\}) \geq 0$.*

The derivations in Theorem 5.6 and Corollary 5.7 generally fit the multi-dimensional latent space (Tumer 1996), such that we can state that compared with the canonical GCL methods, the proposed HTML acquires the relatively tighter Bayes error upper bound, which substantiates that the performance rises of GCL methods incurred by HTML are theoretical solid.

6 Experiments

In this section, we experimentally demonstrate the validity of HTML on 17 real-world datasets and analyze the effects of each module of HTML.

6.1 Experimental Setup

Datasets. For the unsupervised learning task, we conduct experiments on eight regular-sized graphs, I.e., TUDataset (Morris et al. 2020), and one large-scale graph, i.e., Amazon Photo dataset, which has 7,650 nodes and 119,081 edges. For the transfer learning task, we pre-train our model on ZINC-2M (Sterling and Irwin 2015) and finetune it on eight biochemical molecule datasets.

Baselines. We compare HTML with the following baselines: SOTA graph kernel methods such as GL (Shervashidze et al. 2009), WL (Shervashidze et al. 2011), and DGK (Yanardag and Vishwanathan 2015); graph self-supervised learning methods such as InfoGraph (Sun et al. 2020), Random-Init (Velickovic et al. 2019), Infomax (Velickovic et al. 2019), EdgePred (Hu et al. 2020), Attr-Masking (Hu et al. 2020), ContextPred (Hu et al. 2020), and BGRL (Thakoor et al. 2022); graph contrastive learning methods such as GraphCL (You et al. 2020), JOAO (You et al. 2021), AD-GCL (Suresh et al. 2021), RGCL (Li et al. 2022c), PGCL (Lin et al. 2021), MVGRL (Hassani and Ahmadi 2020), and GRACE (Zhu et al. 2020).

6.2 Performance on Real-World Graph Representation Learning Tasks

Benchmarking on unsupervised learning towards regular graphs. Table 1 compares the classification accuracy between our proposed HTML and the SOTA methods under

Methods	NCI1	PROTEINS	DD	MUTAG	COLLAB	RDT-B	RDT-M5K	IMDB-B	AVG.
GL	–	–	–	81.66±2.11	–	77.34±0.18	41.01±0.17	65.87±0.98	66.47
WL	**80.01**±0.50	72.92±0.56	–	80.72±3.00	–	68.82±0.41	46.06±0.21	**72.30**±3.44	70.14
DGK	**80.31**±0.46	73.30±0.82	–	87.44±2.72	–	78.04±0.39	41.27±0.18	66.96±0.56	71.22
InfoGraph	76.20±1.06	74.44±0.31	72.85±1.78	**89.01**±1.13	70.65±1.13	82.50±1.42	53.46±1.03	**73.03**±0.87	74.02
GraphCL	77.87±0.41	74.39±0.45	**78.62**±0.40	86.80±1.34	**71.36**±1.15	89.53±0.84	**55.99**±0.28	71.14±0.44	**75.71**
JOAO	78.07±0.47	**74.55**±0.41	77.32±0.54	87.35±1.02	69.50±0.36	85.29±1.35	55.74±0.63	70.21±3.08	74.75
JOAOv2	78.36±0.53	74.07±1.10	77.40±1.15	87.67±0.79	69.33±0.34	86.42±1.45	**56.03**±0.27	70.83±0.25	75.01
AD-GCL	73.91±0.77	73.28±0.46	75.79±0.87	**88.74**±1.85	**72.02**±0.56	**90.07**±0.85	54.33±0.32	70.21±0.68	74.79
RGCL	78.14±1.08	**75.03**±0.43	**78.86**±0.48	87.66±1.01	70.92±0.65	**90.34**±0.58	**56.38**±0.40	**71.85**±0.84	**76.15**
GraphCL+HTML	**78.72**±0.72	**74.95**±0.33	**78.60**±0.18	**88.91**±1.85	**71.60**±0.81	**90.66**±0.64	55.95±0.36	71.68±0.42	**76.38**
Δ	+0.85	+0.56	-0.02	+2.11	+0.24	+1.13	-0.04	+0.54	+0.67

Table 1: Comparing classification accuracy (%) with unsupervised representation learning setting. We report the average accuracy in the last column, and the top three results are bolded.

Methods	NCI1	MUTAG	COLLAB	IMDB-B	AVG.
No Pre-Train	65.40±0.17	87.39±1.09	65.29±0.16	69.37±0.37	71.86
InfoGraph	76.20±1.06	**89.01**±1.13	70.65±1.13	**73.03**±0.87	77.22
GraphCL	77.87±0.41	86.80±1.34	71.36±1.15	71.14±0.44	76.79
AD-GCL	73.91±0.77	88.74±1.85	**72.02**±0.56	70.21±0.68	76.22
RGCL	78.14±1.08	87.66±1.01	70.92±0.65	71.85±0.84	77.14
RGCL+HTML	**78.38**±0.67	88.72±1.90	71.19±1.13	72.17±0.86	**77.62**
Δ	+0.24	+1.06	+0.27	+0.32	+0.48

Table 2: Comparing classification accuracy (%) with unsupervised representation learning setting. We report the average accuracy in the last column and the improvement compared to the baseline, namely RGCL, in the last row. Bold indicates the best result, and underline indicates the second best result.

the unsupervised representation learning setting. As can be seen, HTML achieves the top three on almost all datasets and ranks first in average accuracy. This verifies our conclusion in **Section** 1, i.e., guiding GNNs to learn topology-level information can improve the capacity of GCL methods to learn discriminative representations. Since HTML is implemented based on GraphCL, we compare the performance between HTML and GraphCL and show the increase in the last row in Table 1. According to Table 1, GraphCL+HTML shows improvements over GraphCL with an average increase of 0.67%.

Benchmarking on unsupervised learning with other baselines. HTML is a plug-and-play method that fits and improves various baselines well. To validate the generality of the performance improvement derived by introducing HTML, we conduct further comparisons on other backbones. We implement HTML with RGCL as the backbone, and the results on four representative datasets are shown in Table 2. The proposed HTML empowers RGCL to achieve the highest average accuracy. Besides, as shown in Figure 3, PGCL+HTML can consistently outperform PGCL over different benchmarks, demonstrating that the improvement brought about by HTML is robust to the variance of the adopted baselines.

Benchmarking on unsupervised learning towards large-scale graphs. To validate the effectiveness of our proposed HTML on large-scale graphs, we implement HTML as a plug-and-play component on BGRL and conduct experiments on the Amazon Photo dataset with 7,650 nodes and 119,081 edges. BGRL is a SOTA model for node classification tasks for unsupervised learning towards large-scale graphs, and the comparisons between BGRL+HTML and benchmarks are shown in Figure 4. Experimental results show that BGRL+HTML achieves the best performance among benchmarks, which validates the generalization of HTML.

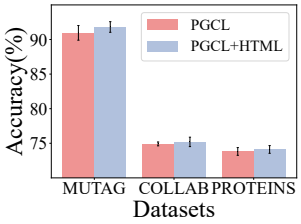

Figure 3: Empirical validation of the improvement robustness of HTML towards the variance of baselines.

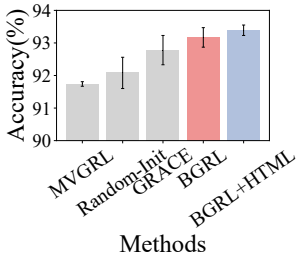

Figure 4: Comparisons on the large-scale Amazon Photos dataset for node classification task.

Benchmarking on transfer learning. Table 3 compares our proposed HTML with the SOTA methods under the transfer learning setting. The last column of this table shows the average improvement compared to the method without pre-training. HTML achieves the highest boost among the compared methods on average, supporting the transferability of our approach. Specifically, we engage in a comparative analysis between HTML and the principal baseline, i.e., GraphCL, in the last row of Table 3. It is worth noting that

Methods	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	HIV	BACE	GAIN.
No Pre-Train	65.8±4.5	74.0±0.8	63.4±0.6	57.3±1.6	58.0±4.4	71.8±2.5	75.3±1.9	70.1±5.4	-
Infomax	68.8±0.8	75.3±0.5	62.7±0.4	58.4±0.8	69.9±3.0	75.3±2.5	76.0±0.7	75.9±1.6	3.3
EdgePred	67.3±2.4	**76.0**±0.6	**64.1**±0.6	60.4±0.7	64.1±3.7	74.1±2.1	76.3±1.0	**79.9**±0.9	3.3
AttrMasking	64.3± 2.8	**76.7**± 0.4	**64.2**± 0.5	**61.0**±0.7	71.8±4.1	74.7±1.4	77.2±1.1	**79.3**±1.6	4.1
ContextPred	68.0±2.0	**75.7**±0.7	**63.9**±0.6	60.9±0.6	65.9±3.8	**75.8**±1.7	77.3±1.0	**79.6**±1.2	3.9
GraphCL	69.68±0.67	73.87±0.66	62.40±0.57	60.53±0.88	75.99±2.65	69.80±2.66	**78.47**±1.22	75.38±1.44	3.77
JOAO	**70.22**±0.98	74.98±0.29	62.94±0.48	59.97±0.79	**81.32**±2.49	71.66±1.43	76.73±1.23	77.34±0.48	**4.90**
AD-GCL	68.26±1.03	73.56±0.72	63.10±0.66	59.24±0.86	77.63±4.21	74.94±2.54	75.45±1.28	75.02±1.88	3.90
RGCL	**71.42**±0.66	75.20±0.34	63.33±0.17	**61.38**±0.61	**83.38**±0.91	**76.66**±0.99	**77.90**±0.80	76.03±0.77	**6.16**
GraphCL+HTML	**70.94**±0.65	74.69±0.28	63.73±0.30	**61.32**±0.59	**84.03**±0.12	**78.05**±0.94	**78.68**±0.61	77.24±0.90	**6.59**
Δ	+1.26	+0.82	+1.33	+0.79	+8.04	+8.25	+0.21	+1.86	+2.82

Table 3: Comparing ROC-AUC scores (%) on downstream graph classification tasks with transfer learning setting.

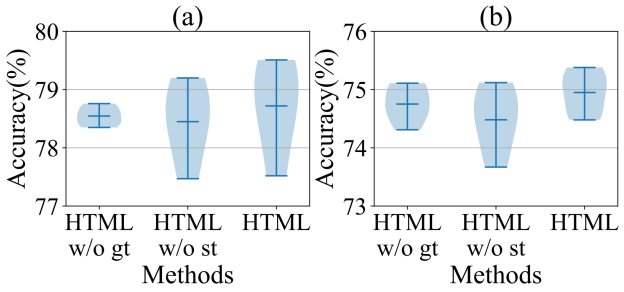

Figure 5: Ablation study on (a) NCI1 and (b) PROTEINS.

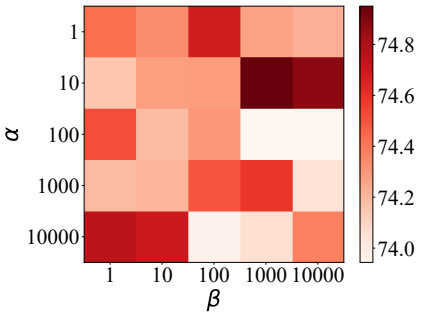

Figure 6: Accuracy (%) with different hyper-parameters.

HTML achieves significant performance boosts in all eight datasets, with an average increase of 2.82%. Compared to the SOTA approach, i.e., RGCL, HTML derives an average performance improvement of 0.43%. Overall, the results demonstrate the valid transferability of HTML.

6.3 In-Depth Analysis and Discussion

Ablation study. To verify the effectiveness of each module in HTML, we conduct ablation studies on NCI1 and PROTEINS datasets, shown in Figure 5. We conduct experiments in three settings: HTML, HTML without graph-tier expertise (HTML w/o gt), and HTML without subgraph-tier expertise (HTML w/o st). We perform five experiments for each setting with five different seeds, and the results are presented as a violin plot. The width of the violin represents the distribution of results, and the blue horizontal line in the middle indicates the mean value of the five results. It can be observed that removing any of the modules from the HTML causes a certain drop in performance. However, the ablation models can still outperform the principle baseline, i.e., GraphCL, which confirms our previous conjecture that GNN-based GCL models lack topology-level information and suggests that our proposed graph-tier expertise and subgraph-tier expertise are both contributing to the GCL model. Figure 5 reveals that adding subgraph-tier expertise leads to a more significant boost to the GCL model and usually has a more stable result under different seeds, indicating that fine-grained expertise is essential for improving model performance and stability.

Hyper-parameter experiment. Appropriate adjustment of the introduction of hierarchical topology isomorphism expertise can better improve the model performance, so we conduct hyperparametric experiments on the PROTEINS dataset for α and β, and the results are shown in Figure 6. α and β are range from $\{1, 1\times10^1, 1\times10^2, 1\times10^3, 1\times10^4\}$. α controls the weight of the graph-tier topology isomorphism expertise module, and β controls the weight of the subgraph-tier topology isomorphism expertise module. As shown in Figure 6, the model achieves the highest accuracy at $\alpha = 10$ and $\beta = 1000$. It is easy to find that when β takes low values, the higher α achieves better results, while when β takes high values, the lower α achieves better results. We argue that this phenomenon may be because the topology isomorphism expertise at graph-tier and subgraph-tier may have certain complementarity.

7 Conclusions

By observing the motivating experiments, we derive the empirical conclusion that GNN-based GCL methods can barely learn the hierarchical topology isomorphism expertise during training. Thus, the corresponding expertise is complementary to GCL methods. To this end, we propose HTML, which introduces the hierarchical topology isomorphism expertise into the GCL paradigm. Theoretically, we provide validated analyses to demonstrate that compared with conventional GCL methods, HTML derives the tighter Bayes error upper bound. Empirically, benchmark comparisons prove that HTML yields significant performance boosts.

Acknowledgements

The authors would like to thank the editors and reviewers for their valuable comments. This work is supported by the 2022 Special Research Assistant Grant project, No. E3YD5901, the CAS Project for Young Scientists in Basic Research, Grant No. YSBR-040, the Youth Innovation Promotion Association CAS, No. 2021106, and the China Postdoctoral Science Foundation, No. 2023M743639.

References

Allen, D. M. 1971. Mean square error of prediction as a criterion for selecting variables. *Technometrics*, 13(3): 469–475.

Babai, L. 2015. Graph Isomorphism in Quasipolynomial Time. *CoRR*, abs/1512.03547.

Babai, L.; and Kucera, L. 1979. Canonical Labelling of Graphs in Linear Average Time. In *20th Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 29-31 October 1979*, 39–46. IEEE Computer Society.

Berne, B. J.; Boon, J. P.; and Rice, S. A. 1966. On the calculation of autocorrelation functions of dynamical variables. *The Journal of Chemical Physics*, 45(4): 1086–1096.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. E. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML 2020, 13-18 July 2020, Virtual Event*, volume 119, 1597–1607. PMLR.

Corneil, D. G.; and Gotlieb, C. C. 1970. An Efficient Algorithm for Graph Isomorphism. *J. ACM*, 17(1): 51–64.

De Maesschalck, R.; Jouan-Rimbaud, D.; and Massart, D. L. 2000. The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1): 1–18.

Gao, H.; Li, J.; Qiang, W.; Si, L.; Xu, B.; Zheng, C.; and Sun, F. 2023. Robust causal graph representation learning against confounding effects. In *AAAI*, volume 37, 7624–7632.

Gao, T.; Yao, X.; and Chen, D. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In Moens, M.; Huang, X.; Specia, L.; and Yih, S. W., eds., *EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, 6894–6910. Association for Computational Linguistics.

Grill, J.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. Á.; Guo, Z.; Azar, M. G.; Piot, B.; Kavukcuoglu, K.; Munos, R.; and Valko, M. 2020. Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *NeurIPS 2020, December 6-12, 2020, virtual*.

Hamilton, W. L.; Ying, Z.; and Leskovec, J. 2017. Inductive Representation Learning on Large Graphs. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *NeurIPS 2017, December 4-9, 2017, Long Beach, CA, USA*, 1024–1034.

Hassani, K.; and Ahmadi, A. H. K. 2020. Contrastive Multi-View Representation Learning on Graphs. In *ICML 2020, 13-18 July 2020, Virtual Event*, volume 119, 4116–4126. PMLR.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. B. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 9726–9735. Computer Vision Foundation / IEEE.

Heckman, J. J. 1979. Sample selection bias as a specification error. *Econometrica: Journal of the econometric society*, 153–161.

Hu, W.; Liu, B.; Gomes, J.; Zitnik, M.; Liang, P.; Pande, V. S.; and Leskovec, J. 2020. Strategies for Pre-training Graph Neural Networks. In *8th ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jaccard, P. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2): 37–50.

Jain, A. K.; Duin, R. P. W.; and Mao, J. 2000. Statistical Pattern Recognition: A Review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(1): 4–37.

Jiao, Y.; Xiong, Y.; Zhang, J.; Zhang, Y.; Zhang, T.; and Zhu, Y. 2020. Sub-graph Contrast for Scalable Self-Supervised Graph Representation Learning. In Plant, C.; Wang, H.; Cuzzocrea, A.; Zaniolo, C.; and Wu, X., eds., *ICDM 2020, Sorrento, Italy, November 17-20, 2020*, 222–231. IEEE.

Ju, W.; Gu, Y.; Luo, X.; Wang, Y.; Yuan, H.; Zhong, H.; and Zhang, M. 2023. Unsupervised graph-level representation learning with hierarchical contrasts. *Neural Networks*, 158: 359–368.

Kipf, T. N.; and Welling, M. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *CoRR*, abs/1609.02907.

Li, J.; Qiang, W.; Zhang, Y.; Mo, W.; Zheng, C.; Su, B.; and Xiong, H. 2022a. MetaMask: Revisiting Dimensional Confounder for Self-Supervised Learning. *NeurIPS*, 35: 38501–38515.

Li, J.; Qiang, W.; Zheng, C.; Su, B.; and Xiong, H. 2022b. Metaug: Contrastive learning via meta feature augmentation. In *ICML 2022*, 12964–12978. PMLR.

Li, S.; Wang, X.; Zhang, A.; Wu, Y.; He, X.; and Chua, T. 2022c. Let Invariant Rationale Discovery Inspire Graph Contrastive Learning. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári, C.; Niu, G.; and Sabato, S., eds., *ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162, 13052–13065. PMLR.

Lin, L.; Chen, J.; and Wang, H. 2023. Spectral Augmentation for Self-Supervised Learning on Graphs. In *ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Lin, S.; Zhou, P.; Hu, Z.; Wang, S.; Zhao, R.; Zheng, Y.; Lin, L.; Xing, E. P.; and Liang, X. 2021. Prototypical Graph Contrastive Learning. *CoRR*, abs/2106.09645.

McKay, B. D.; and Piperno, A. 2014. Practical graph isomorphism, II. *J. Symb. Comput.*, 60: 94–112.

Morris, C.; Kriege, N. M.; Bause, F.; Kersting, K.; Mutzel, P.; and Neumann, M. 2020. TUDataset: A collection of benchmark datasets for learning with graphs. *CoRR*, abs/2007.08663.

Qiang, W.; Li, J.; Zheng, C.; Su, B.; and Xiong, H. 2022. Interventional Contrastive Learning with Meta Semantic Regularizer. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári,

C.; Niu, G.; and Sabato, S., eds., *ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162, 18018–18030. PMLR.

Qu, Y.; Shen, D.; Shen, Y.; Sajeev, S.; Chen, W.; and Han, J. 2021. CoDA: Contrast-enhanced and Diversity-promoting Data Augmentation for Natural Language Understanding. In *ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Rajaraman, S.; Zamzmi, G.; and Antani, S. K. 2021. Multi-loss ensemble deep learning for chest X-ray classification. *CoRR*, abs/2109.14433.

Rosenblatt, M. 1956. A central limit theorem and a strong mixing condition. *Proceedings of the national Academy of Sciences*, 42(1): 43–47.

Shervashidze, N.; Schweitzer, P.; van Leeuwen, E. J.; Mehlhorn, K.; and Borgwardt, K. M. 2011. Weisfeiler-Lehman Graph Kernels. *J. Mach. Learn. Res.*, 12: 2539–2561.

Shervashidze, N.; Vishwanathan, S. V. N.; Petri, T.; Mehlhorn, K.; and Borgwardt, K. M. 2009. Efficient graphlet kernels for large graph comparison. In Dyk, D. A. V.; and Welling, M., eds., *AISTATS 2009, Clearwater Beach, Florida, USA, April 16-18, 2009*, volume 5, 488–495. JMLR.org.

Shimodaira, H. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2): 227–244.

Sohn, K. 2016. Improved Deep Metric Learning with Multi-class N-pair Loss Objective. In Lee, D. D.; Sugiyama, M.; von Luxburg, U.; Guyon, I.; and Garnett, R., eds., *NeurIPS 2016, December 5-10, 2016, Barcelona, Spain*, 1849–1857.

Sterling, T.; and Irwin, J. J. 2015. ZINC 15 - Ligand Discovery for Everyone. *J. Chem. Inf. Model.*, 55(11): 2324–2337.

Sun, F.; Hoffmann, J.; Verma, V.; and Tang, J. 2020. InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization. In *ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Suresh, S.; Li, P.; Hao, C.; and Neville, J. 2021. Adversarial graph augmentation to improve graph contrastive learning. *NeurIPS*, 34: 15920–15933.

Tang, H.; Zhang, X.; Wang, J.; Cheng, N.; and Xiao, J. 2022. Avqvc: One-Shot Voice Conversion By Vector Quantization With Applying Contrastive Learning. In *ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, 4613–4617. IEEE.

Thakoor, S.; Tallec, C.; Azar, M. G.; Azabou, M.; Dyer, E. L.; Munos, R.; Velickovic, P.; and Valko, M. 2022. Large-Scale Representation Learning on Graphs via Bootstrapping. In *ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Tumer, K. 1996. *Linear and order statistics combiners for reliable pattern classification*. The University of Texas at Austin.

van den Oord, A.; Li, Y.; and Vinyals, O. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR*, abs/1807.03748.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2017. Graph Attention Networks. *CoRR*, abs/1710.10903.

Velickovic, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep Graph Infomax. In *ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Weisfeiler, B.; and Leman, A. 1968. The reduction of a graph to canonical form and the algebra which appears therein. *nti, Series*, 2(9): 12–16.

Wijesinghe, A.; and Wang, Q. 2022. A New Perspective on "How Graph Neural Networks Go Beyond Weisfeiler-Lehman?". In *ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Wu, Z.; Xiong, Y.; Yu, S. X.; and Lin, D. 2018. Unsupervised Feature Learning via Non-Parametric Instance Discrimination. In *CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 3733–3742. Computer Vision Foundation / IEEE Computer Society.

Xia, P.; Zhang, L.; and Li, F. 2015. Learning similarity with cosine similarity ensemble. *Inf. Sci.*, 307: 39–52.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How Powerful are Graph Neural Networks? *CoRR*, abs/1810.00826.

Yanardag, P.; and Vishwanathan, S. V. N. 2015. Deep Graph Kernels. In Cao, L.; Zhang, C.; Joachims, T.; Webb, G. I.; Margineantu, D. D.; and Williams, G., eds., *SIGKDD, Sydney, NSW, Australia, August 10-13, 2015*, 1365–1374. ACM.

Yao, D.; Zhao, Z.; Zhang, S.; Zhu, J.; Zhu, Y.; Zhang, R.; and He, X. 2022. Contrastive Learning with Positive-Negative Frame Mask for Music Representation. In Laforest, F.; Troncy, R.; Simperl, E.; Agarwal, D.; Gionis, A.; Herman, I.; and Médini, L., eds., *WWW 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, 2906–2915. ACM.

You, Y.; Chen, T.; Shen, Y.; and Wang, Z. 2021. Graph Contrastive Learning Automated. In Meila, M.; and Zhang, T., eds., *ICML 2021, 18-24 July 2021, Virtual Event*, volume 139, 12121–12132. PMLR.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph Contrastive Learning with Augmentations. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *NeurIPS 2020, December 6-12, 2020, virtual*.

Zbontar, J.; Jing, L.; Misra, I.; LeCun, Y.; and Deny, S. 2021. Barlow Twins: Self-Supervised Learning via Redundancy Reduction. In Meila, M.; and Zhang, T., eds., *ICML 2021, 18-24 July 2021, Virtual Event*, volume 139, 12310–12320. PMLR.

Zhang, S.; Hu, Z.; Subramonian, A.; and Sun, Y. 2020. Motif-Driven Contrastive Learning of Graph Representations. *CoRR*, abs/2012.12533.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2020. Deep Graph Contrastive Representation Learning. *CoRR*, abs/2006.04131.