# PRISM: PARTIAL-LABEL RELATIONAL INFERENCE WITH SPATIAL AND SPECTRAL CUES

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

In many real-world scenarios, precisely labeling graph data is costly or impractical, especially in domains like molecular biology or social networks, where annotation requires expert effort. This challenge motivates partial-label graph learning, where each graph is weakly annotated with a candidate label set containing the true label. However, such ambiguous supervision makes it hard to extract reliable semantics and increases the risk of overfitting to noisy candidates. To address these challenges, we propose PRISM, a unified framework that performs relational inference with spatial and spectral cues to resolve label ambiguity. PRISM captures discriminative spatial cues by aligning prototype-guided substructures across graphs and extracts global spectral cues by decomposing graph signals into multiple frequency bands with attention, preserving frequency-specific semantics. These complementary views are integrated into a hybrid relational graph, which supports confidence-aware label propagation under candidate constraints. A closed-loop refinement mechanism further stabilizes supervision via masked updates and momentum-based confidence estimation. Extensive experiments across diverse benchmarks demonstrate that PRISM consistently outperforms strong baselines under various noise settings, establishing a new paradigm for weakly supervised graph classification. The source code is available at https://anonymous.4open.science/r/PRISM-17107/.

## 1 INTRODUCTION

Graph-structured data is pervasive across diverse domains such as drug discovery, molecular property prediction, social network analysis, and recommendation systems Fang et al. (2022); Zhang et al. (2021b); Wang et al. (2021). These data are naturally represented as graphs, where nodes denote entities and edges model relations. To handle such complex structures, Graph Neural Networks (GNNs) Welling & Kipf (2016); Hamilton et al. (2017); Xu et al. (2018); Zhang et al. (2021a) have emerged as powerful tools for learning expressive graph-level representations, achieving state-of-the-art results in a wide range of applications including biomedical classification Liu et al. (2023), cross-modal retrieval Chen et al. (2022), and event understanding Du et al. (2023). GNNs typically operate by recursively aggregating information from node neighborhoods and summarizing node embeddings via global pooling Gao & Ji (2019); Lee et al. (2021), followed by classification.

Despite these advances, existing GNN-based graph classification frameworks are fundamentally data-intensive: they require accurate and fully labeled training samples to learn discriminative representations Li et al. (2022); Rousseau et al. (2015). However, in many real-world scenarios, acquiring ground-truth labels is prohibitively expensive or technically infeasible. For instance, annotating molecular graphs often depends on density functional theory (DFT) simulations Becke (2014), which are computationally demanding; in social or biomedical networks, labeling requires domain experts to manually verify latent properties such as toxicity, protein function, or community membership Yu et al. (2015a). As a result, datasets in practice are often weakly supervised, with only partial or ambiguous label information available. While self-supervised methods such as GraphCL You et al. (2020) alleviate reliance on labels during pre-training by leveraging contrastive objectives, they still depend on accurate annotations during fine-tuning or evaluation stages. In the presence of label ambiguity, their performance tends to degrade significantly due to misaligned supervision signals. This raises a crucial question: *how can we train reliable GNN classifiers in the presence of incomplete or uncertain supervision?*

We focus on a practical yet underexplored setting: Partial-label Graph Learning (PLGL). Each graph is annotated with a candidate label set $\mathcal{S}_i \subset \mathcal{Y}$ containing the true class label, but the exact ground-truth is unknown. This situation commonly arises when annotations are generated from noisy heuristics, coarse rules, or automated labeling tools Ge et al. (2022). While similar ideas have been studied in image classification Feng & An (2019); Lyu et al. (2020), PLGL presents unique challenges due to the structural complexity and non-Euclidean nature of graphs. First, ambiguous supervision introduces semantic uncertainty, making it difficult to capture class-discriminative substructures. Second, standard GNNs tend to overfit noisy signals without precise feedback, particularly when candidate sets contain semantically similar labels. Third, unlike images, graphs exhibit patterns at multiple structural resolutions, from local motifs to global topology, which cannot be represented by uniform pooling or global averaging. Recent works have explored weakly supervised graph classification using pseudo-labeling Ju et al. (2023) or contrastive learning You et al. (2020); Luo et al. (2023), but they often face two limitations: (i) reliance on single-view or overconfident predictions, leading to error accumulation and limited robustness under severe ambiguity; and (ii) lack of explicit use of structural and spectral diversity to disentangle candidate labels.

To bridge this gap, we introduce PRISM, a unified and theoretically grounded framework for Partial-label Relational Inference with Spatial and Spectral Modeling. PRISM tackles label ambiguity through three synergistic and mutually reinforcing perspectives. First, it extracts *spatial cues* by aligning prototype-guided substructures across graphs, thereby uncovering class-discriminative local patterns even under noisy candidate sets. Second, it encodes *spectral cues* by decomposing graph signals into multiple frequency bands, where a dedicated multi-band attention mechanism preserves frequency-specific semantics critical for fine-grained reasoning and global structural understanding. Third, it constructs a *hybrid relational graph* that integrates both spatial and spectral similarities, enabling confidence-aware label propagation to refine supervision signals iteratively and coherently. A momentum-based update of soft labels under candidate constraints forms a closed-loop refinement mechanism, which stabilizes training, suppresses noise accumulation, and ultimately yields robust and reliable optimization under weak supervision.

Our contributions are summarized as follows. (1) **Underexplored Problem.** We study the underexplored problem of partial-label graph learning, motivated by practical scenarios with ambiguous supervision in molecular, social, and multimedia domains. (2) **Novel Framework.** We propose PRISM, a novel relational inference framework that integrates spatial and spectral cues to disambiguate labels, combining substructure alignment, frequency-aware encoding, and dual-relational propagation. (3) **Extensive experiments.** We validate PRISM across diverse benchmarks and demonstrate its superiority over existing weakly supervised and graph learning approaches.

## 2 Background

**Problem Definition.** Let $\mathcal{G} = \{G_i = (\mathcal{V}_i, \mathcal{E}_i, \boldsymbol{X}_i)\}_{i=1}^{N}$ denote a collection of $N$ graphs, where each graph $G_i$ consists of a node set $\mathcal{V}_i$, edge set $\mathcal{E}_i$, and node features $\boldsymbol{X}_i \in \mathbb{R}^{|\mathcal{V}_i| \times d}$. We denote by $\boldsymbol{A}_i \in \{0,1\}^{|\mathcal{V}_i| \times |\mathcal{V}_i|}$ the adjacency matrix of $\mathcal{E}_i$. For each graph $G_i$, we are given a candidate label set $\mathcal{S}_i \subset \mathcal{Y}$, where $\mathcal{Y} = \{1, 2, \ldots, C\}$ is the complete label space. The candidate set $\mathcal{S}_i$ includes the true label $y_i^*$ but does not reveal which one is correct. Our objective is to learn a graph classifier $f(G_i; \theta)$ that predicts the ground-truth label $y_i^*$ for each graph in the test set, by training only on ambiguous candidate sets without access to ground-truth supervision.

**Graph Neural Networks.** Graph Neural Networks (GNNs) are widely used to encode graph structures by recursively aggregating information from node neighborhoods. At each layer $l$, the representation of node $v$ is updated by combining its own embedding with messages from its neighbors:

$$\boldsymbol{h}_v^{(l)} = \phi^{(l)} \left( \boldsymbol{h}_v^{(l-1)}, \sum_{u \in \mathcal{N}(v)} \psi^{(l)} \left( \boldsymbol{h}_u^{(l-1)} \right) \right), \tag{1}$$

where $\phi^{(l)}$ and $\psi^{(l)}$ are learnable functions, and $\mathcal{N}(v)$ denotes the neighbor set of node $v$.

After $L$ layers, node embeddings $\boldsymbol{h}_v^{(L)}$ are aggregated into a graph-level representation using a readout function:

$$\boldsymbol{g} = \text{READOUT} \left( \{\boldsymbol{h}_v^{(L)}\}_{v \in \mathcal{V}} \right), \tag{2}$$

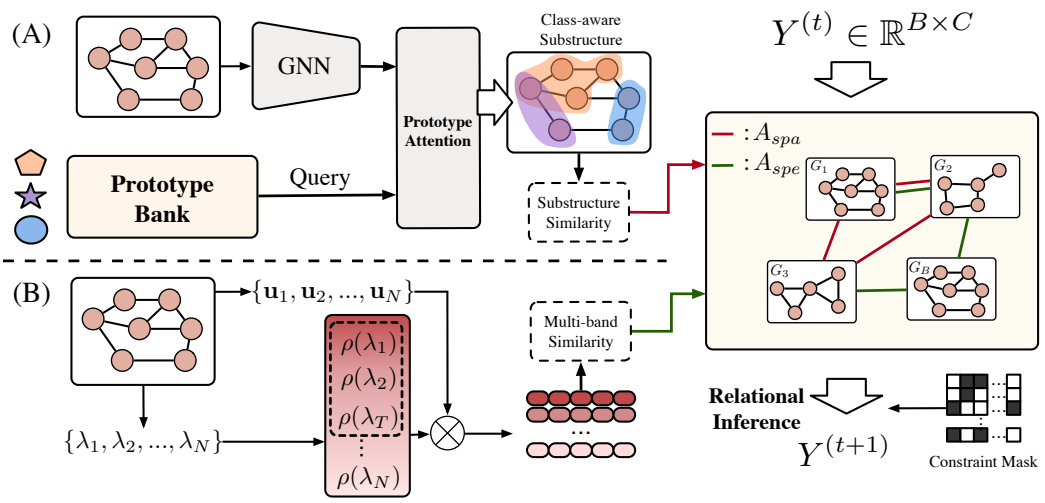



Figure 1: Overview of PRISM. Our framework jointly models spatial substructures (A) and spectral frequency patterns (B) to disambiguate partial labels. Prototype-guided attention and multi-band spectral encoding construct two relational graphs ($\boldsymbol{A}^{spa}$, $\boldsymbol{A}^{spe}$) for iterative label refinement under candidate constraints.

where READOUT can be sum, mean, or attention-based pooling. These graph representations provide the basis for downstream tasks such as graph-level classification.

## 3 Methodology

This paper introduces a novel framework PRISM for partial-label graph learning that integrates local substructural cues and global spectral dynamics to resolve candidate label ambiguities. Each graph is simultaneously encoded through two complementary pathways: a spatial encoder emphasizes discriminative regions by aligning interpretable subgraphs with label prototypes, while a spectral encoder decomposes graph signals into distinct frequency bands, capturing both smooth and irregular structural patterns. These dual perspectives induce a relational graph with two types of edges: one encoding prototype-based substructure similarity, the other reflecting spectral affinity across graphs. To refine label supervision, we perform confidence-aware propagation over this hybrid graph, fusing signals from both relational types. This design enables the model to distill consistent label cues from noisy candidates, without relying on ground-truth labels. The overview of the proposed framework PRISM is illustrated in Figure 1, and we will elaborate on the details below.

### 3.1 Spatial Cues via Substructure Matching

In partial-label graph learning, the true label of each sample is concealed within a noisy candidate set, where multiple labels may be semantically correlated. Relying solely on global representations often blurs these distinctions, since graphs with overlapping candidate labels can exhibit similar overall topology. In contrast, local substructures frequently encode the most discriminative evidence for class separation. Motivated by this observation, we design a structure-aware disambiguation module that aligns interpretable subgraph-level components across related instances, enabling prototype-guided reasoning to uncover consistent label semantics under ambiguity.

To incorporate class-level semantic priors, we maintain a momentum-updated *prototype bank* $\{\boldsymbol{p}_c \in \mathbb{R}^F\}_{c\in\mathcal{Y}}$, where each prototype $\boldsymbol{p}_c$ tracks the aggregated global representation of graphs associated with candidate label $c$. At training step $t$, we update each prototype as:

$$\boldsymbol{p}_c^{(t)} \leftarrow m \cdot \boldsymbol{p}_c^{(t-1)} + (1-m) \cdot \frac{1}{|\mathcal{B}_c|} \sum_{i\in\mathcal{B}_c} \boldsymbol{g}_i, \tag{3}$$

where $\boldsymbol{g}_i = \text{READOUT}\left(\{\boldsymbol{h}_v^{(L)}\}_{v\in\mathcal{V}_i}\right)$ denotes the global embedding of graph $G_i$, and $\mathcal{B}_c$ is the set of current-batch graphs containing label $c$ in their candidate set and passing confidence-based filter-

ing. Based on the prototype bank, we apply a *prototype-guided attention mechanism* to extract $C$ substructure embeddings per graph, where each embedding is aligned with a candidate class. Given the node embeddings from the final GNN layer $\{\boldsymbol{h}_v^{(L)}\}_{v\in\mathcal{V}_i}$ and the full prototype set $\{\boldsymbol{p}_c\}_{c=1}^{C}$, attention weights are computed to obtain class-aware latent components:

$$\boldsymbol{r}_i^{(c)} = \sum_{v\in\mathcal{V}_i} \alpha_{vc} \cdot \boldsymbol{h}_v^{(L)}, \text{where } \alpha_{vc} = \frac{\exp(\boldsymbol{h}_v^{(L)\top}\boldsymbol{p}_c)}{\sum_{v'}\exp(\boldsymbol{h}_{v'}^{(L)\top}\boldsymbol{p}_c)}. \tag{4}$$

The resulting embeddings $\{\boldsymbol{r}_i^{(c)}\}_{c=1}^{C}$ serve as interpretable, class-specific substructures for downstream comparison. We then construct a relational graph over graph pairs that share at least one candidate label. Let $\mathcal{P} = \{(i,j) \mid \mathcal{S}_i \cap \mathcal{S}_j \neq \emptyset,\ i \neq j\}$ denote the set of such graph pairs. For each pair $(G_i, G_j) \in \mathcal{P}$, we define a prototype-aware substructure similarity:

$$s_{ij}^{spa} = \max_{c\in\mathcal{S}_i\cap\mathcal{S}_j} \cos\left(\boldsymbol{r}_i^{(c)}, \boldsymbol{r}_j^{(c)}\right) \cdot \cos\left(\frac{\boldsymbol{r}_i^{(c)} + \boldsymbol{r}_j^{(c)}}{2}, \boldsymbol{p}_c\right). \tag{5}$$

For each graph, we retain its top-$k_a$ neighbors with the highest $s_{ij}^{spa}$ scores to form a sparse relational graph with a normalized adjacency matrix $\boldsymbol{A}^{spa}$ that encodes substructure-level agreement under label semantics. This spatial reasoning module provides a fine-grained structural prior, thereby enhancing label disambiguation by promoting relational consistency among structurally aligned and semantically plausible graph instances across diverse scenarios.

### 3.2 Spectral Cues via Multi-Band Frequency Attention

While the spatial disambiguation module focuses on extracting local spatial cues, graph spectra offer a complementary global perspective by capturing both low-frequency smoothness and high-frequency irregularities. However, many existing spectral methods treat all frequency components equally, applying uniform aggregation across the spectrum. This equal weighting tends to blur structurally diverse signals and may obscure frequency-specific patterns that are crucial for fine-grained graph understanding. In contrast, we propose a *Multi-Band Frequency Attention* module that not only integrates spectral information into a unified representation but also preserves frequency-specific characteristics through explicit band-wise modeling. Specifically, our method: (1) maintains the resolution of each frequency band by employing independently parameterized encoders, and (2) supports fine-grained cross-graph reasoning via band-level similarity comparison. This design enables frequency-aware graph embeddings that emphasize informative spectral patterns while suppressing noise from less relevant bands.

Given a graph $G = (\mathcal{V}, \mathcal{E})$ with normalized Laplacian eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ and corresponding eigenvectors $\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_N\}$, we first lift scalar spectral values into a learnable signal space using harmonic expansion:

$$\rho(\lambda) = [\sin(k\lambda), \cos(k\lambda)]_{k=1}^{T} \cdot \boldsymbol{W}_\rho, \quad \rho(\lambda) \in \mathbb{R}^d, \tag{6}$$

where $\boldsymbol{W}_\rho \in \mathbb{R}^{2T\times d}$ is a shared learnable projection matrix. This produces $T$ distinct frequency embeddings, each serving as a filtered spectral descriptor of the graph. To construct band-specific node representations, we modulate the $p$-th eigenvector $\boldsymbol{u}_p \in \mathbb{R}^N$ with its associated harmonic encoding $\rho(\lambda_p)$ to form:

$$\boldsymbol{X}^{(p)} = \boldsymbol{u}_p \otimes \rho(\lambda_p) \in \mathbb{R}^{N\times d}, \tag{7}$$

where $\otimes$ denotes the outer product broadcast across all nodes. Each $\boldsymbol{X}^{(p)}$ is then processed by an independently parameterized feedforward network (MLP), resulting in transformed node features $\tilde{\boldsymbol{X}}^{(p)} \in \mathbb{R}^{N\times d}$ that encode frequency-specific semantics. We then apply a message passing neural network $f_{\text{shared}}$, shared across all frequency bands, to each $\tilde{\boldsymbol{X}}^{(p)}$ to extract high-level structural signals. For each band $p \in \{1, \ldots, T\}$, we compute the corresponding graph-level embedding as:

$$\boldsymbol{z}^{(p)} = \text{READOUT}\left(f_{\text{shared}}(\tilde{\boldsymbol{X}}^{(p)}, \boldsymbol{A})\right), \tag{8}$$

where $\boldsymbol{A}$ denotes the adjacency matrix of the graph. This yields a set of band-level embeddings $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(T)}\}$ that capture structurally filtered representations of the graph across multiple fre-

quency perspectives. To synthesize multi-scale structural signals, we employ a soft attention mechanism across bands:

$$\boldsymbol{z} = \sum_{p=1}^{T} \alpha^{(p)} \boldsymbol{z}^{(p)}, \text{where } \alpha^{(p)} = \frac{\exp\left(\boldsymbol{a}^\top \sigma(\boldsymbol{W}\boldsymbol{z}^{(p)})\right)}{\sum_{q=1}^{T} \exp\left(\boldsymbol{a}^\top \sigma(\boldsymbol{W}\boldsymbol{z}^{(q)})\right)}. \tag{9}$$

Here, $\boldsymbol{W}$ and $\boldsymbol{a}$ are learnable parameters, and $\sigma(\cdot)$ denotes a nonlinear activation. For cross-graph structural reasoning, we compute band-wise similarity between graphs. Let $G_i$ and $G_j$ denote two graphs with band embeddings $\{\boldsymbol{z}_i^{(p)}\}_{p=1}^{T}$ and $\{\boldsymbol{z}_j^{(p)}\}_{p=1}^{T}$, respectively. Their similarity is defined as:

$$s_{ij}^{spe} = \max_{p \in \{1,\ldots,T\}} \cos\left(\boldsymbol{z}_i^{(p)}, \boldsymbol{z}_j^{(p)}\right). \tag{10}$$

To ensure label-aware alignment, we restrict edges to graph pairs with overlapping candidate sets, i.e., $\mathcal{S}_i \cap \mathcal{S}_j \neq \emptyset$. Each graph links to its top-$k_e$ most similar neighbors, forming a relational graph with normalized adjacency matrix $\boldsymbol{A}^{spe}$ that supports label disambiguation.

We theoretically establish that both $\boldsymbol{A}^{spa}$ and $\boldsymbol{A}^{spe}$ contribute to label disambiguation. Intuitively, the distribution of hidden node embeddings should be determined by the prototype of the true label to a certain degree. Building on this observation, we present the following theorem with proof provided in Appendix A:

**Theorem 1.** *Assume* $\mathbb{E}\left[h_v^{(i)} | y_i^* = c\right] = p_c$, $\forall v \in \mathcal{V}_i$, $\eta_{jk}^{(i)} = \mathbb{P}\left(\boldsymbol{A}_i\left(j,k\right) = 1\right)$ *is random variable whose distribution can be determined by* $y_i^*$, *and* $\min_{1 \le i \le N, 1 \le p \le T} \left|\lambda_{p+1}^{(i)} - \lambda_p^{(i)}\right| \ge \delta$ *for some* $\delta > 0$. *Then, we have for any* $i, j \in \{1, 2, ..., N\}$*:*

$$\mathbb{P}\left(A_{ij}^{spa} = 1 | y_i^* = y_j^*\right) \to 1 \tag{11}$$

*and*

$$\mathbb{P}\left(A_{ij}^{spe} = 1 | y_i^* = y_j^*\right) \to 1 \tag{12}$$

*as* $\left|\mathcal{V}_i\right|, \left|\mathcal{V}_j\right| \to \infty$.

Theorem 1 suggests that if graph $i$ and $j$ shares the same true label, the probabilities of $A_{ij}^{spa}$ and $A_{ij}^{spe}$ being 1 will both converge to 1, which works on the subsequent label propagation with disambiguation from noisy candidate sets.

### 3.3 Label Disambiguation via Relational Inference

Based on the relational graph constructed in previous modules, which encodes spatial proximity and spectral correlation as distinct relational types, we develop an iterative label propagation framework to enhance supervision under partially labeled settings where ground-truth labels are inaccessible. This framework integrates complementary structural and spectral signals, progressively refining soft supervision, while rigorously enforcing candidate label constraints throughout the entire refinement process to ensure both semantic validity and training stability.

Let $\boldsymbol{Y}^{(0)} \in \mathbb{R}^{N \times C}$ denote the initial soft label matrix, where $N$ is the number of graphs and $C$ the number of classes. At each iteration $t$, label signals are updated using two normalized adjacency matrices: $\boldsymbol{A}^{spa}$ for spatial relations and $\boldsymbol{A}^{spe}$ for spectral affinity. The propagation rule is:

$$\tilde{\boldsymbol{Y}}^{(t+1)} = \alpha \cdot \boldsymbol{Y}^{(t)} + (1-\alpha) \cdot \mathcal{N}\left(\boldsymbol{A}^{spa}\boldsymbol{Y}^{(t)} + \boldsymbol{A}^{spe}\boldsymbol{Y}^{(t)}\right), \tag{13}$$

where $\alpha \in (0,1)$ controls the update momentum, and $\mathcal{N}(\cdot)$ denotes row-wise $\ell_1$ normalization. To constrain label propagation to valid candidate classes, we apply a binary mask $\boldsymbol{M} \in \{0,1\}^{N \times C}$ after each update:

$$\boldsymbol{Y}^{(t+1)} = \mathcal{N}\left(\tilde{\boldsymbol{Y}}^{(t+1)} \odot \boldsymbol{M}\right), \tag{14}$$

where $\odot$ denotes element-wise multiplication. After $T$ iterations, the refined label matrix $\boldsymbol{Y}^{(T)}$ captures multi-relational consistency while remaining faithful to partial supervision, enabling effective disambiguation of noisy candidate sets. We maintain a soft label confidence matrix $\boldsymbol{Q} \in \mathbb{R}^{N \times C}$,

initialized uniformly over candidate classes. To adaptively improve supervision quality, $\boldsymbol{Q}$ is periodically updated using the soft labels $\boldsymbol{Y}^{(T)}$ inferred through relational propagation. The update follows an exponential moving average (EMA) scheme:

$$\boldsymbol{Q}_i \leftarrow \mathcal{N}\left(m \cdot \boldsymbol{Q}_i + (1-m) \cdot \boldsymbol{Y}_i^{(T)}\right), \tag{15}$$

where $m \in (0,1)$ is the momentum coefficient and $\mathcal{N}(\cdot)$ denotes row-wise $\ell_1$ normalization applied over candidate entries. This closed-loop mechanism ensures alignment between model predictions and structure-aware label signals, promoting stable and reliable supervision during training.

### 3.4 Unified Training Objective

To train the model under partial supervision, we adopt a unified objective that couples candidate-constrained loss with confidence-aware refinement. The final predictions are obtained by applying an MLP-based classifier to the spatial-view embeddings $\boldsymbol{g}$, producing logits $\boldsymbol{P}^{spa} \in \mathbb{R}^{N\times C}$. In parallel, a distinct classifier processes the spectral-view embeddings $\boldsymbol{z}$ to yield $\boldsymbol{P}^{spe} \in \mathbb{R}^{N\times C}$. The training loss for the spatial (spectral) view is defined as the negative marginal log-likelihood over candidate classes:

$$\mathcal{L}_{sup}^{(o)} = -\frac{1}{B}\sum_{i=1}^{B} \log \sum_{c\in\mathcal{S}_i} \text{Softmax}(\boldsymbol{P}_i^{(o)})_c \cdot \boldsymbol{Q}_{ic}, \quad o \in \{spa, spe\}, \tag{16}$$

where $\mathcal{S}_i$ is the candidate label set for sample $i$, and $B$ is the batch size. This formulation encourages the model to align predictions with the confidence-weighted support within each candidate set. We jointly optimize the spatial and spectral objectives to extract complementary supervisory signals:

$$\mathcal{L} = \mathcal{L}_{sup}^{spa} + \mathcal{L}_{sup}^{spe}, \tag{17}$$

This dual-view supervision facilitates robust label disambiguation under uncertainty. We then offer a theoretical analysis of the proposed method, particularly focusing on the convergence of the label confidence matrix and training loss under certain conditions. To begin with, let $Y^* = (Y_i^*)_{i=1}^N \in \{0,1\}^{N\times C}$ be the matrix consisting of ground-truth one-hot label vector. Denote the classifier as $f_{classifier}$ which produces partial label confidence matrix $P$, the final predicted label confidence vector of graph $i$ is $P_i = f_{classifier}(g_i)$. If the classifier is well-trained, it should recover label $c$ from $p_c$ since $p_c$ is the prototype of label $c$. Based on this, we have the following results with the proof in Appendix B:

**Theorem 2.** *Under the assumption of Theorem 1, further assume $f_{classifier}(p_c) = \mathbb{I}_c,\ \forall c \in \mathcal{Y}$ where $\mathbb{I}_c \in \{0,1\}^C$ denotes one-hot vector whose c-th component is 1 while the rest are 0, we have:*

$$Q_i \xrightarrow{a.s.} Y_i^* \tag{18}$$

*as $|\mathcal{V}_i|, T \to \infty$. And*

$$\mathbb{E}[\mathcal{L}_{sup}] \to 0 \tag{19}$$

*as $\min_{1\le i\le N} |\mathcal{V}_i|, T \to \infty$.*

Theorem 2 indicates that if each graph has enough node information and we iterate enough epochs, the soft label confidence matrix updated by EMA will converge to the ground-truth and training loss will tend to zero, which further implies our framework can resolve label ambiguity by aligning prototype-guided substructures across graphs. The condition $|\mathcal{V}_i| \to \infty$ can be replaced by $B, N \to \infty$ to a certain degree, since graphs with the same label and candidate label set can be regarded as a whole, and the whole number of nodes tends to infinity when $B, N \to \infty$.

### 3.5 Computational Efficiency Analysis

Let $N$ be the number of nodes, $|\mathcal{E}|$ the number of edges, $d$ the feature dimension, $L$ the number of GNN layers, and $T$ the number of spectral bands. In preprocessing, we compute the $k$ smallest eigenvectors and their spectral encodings, which are reused throughout training. During training, the spatial view performs message passing with complexity $\mathcal{O}(L|\mathcal{E}|d)$, while the spectral view operates on pre-computed features across $T$ bands and applies shared MLPs, resulting in a total cost of $\mathcal{O}(TNd)$. Since $L$ and $T$ are small constants, the overall training complexity is $\mathcal{O}(|\mathcal{E}|d)$, which is linear in the number of edges and consistent with standard GNN-based methods.

Table 1: The classification accuracy (mean%±std%) on five graph benchmark datasets. The best results are shown in boldface and the second best results are underlined. $q = P(\overline{y} \in Y | \overline{y} \neq y)$ reflecting the degree of label ambiguity.

| Datasets | ENZYMES | | Letter-High | | COIL-DEL | | CIFAR10 | | COLORS-3 | |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | $q=0.3$ | $q=0.5$ | $q=0.3$ | $q=0.5$ | $q=0.05$ | $q=0.1$ | $q=0.3$ | $q=0.5$ | $q=0.3$ | $q=0.5$ |
| GCN | 48.44±2.06 | 40.22±2.93 | 44.00±1.08 | 35.94±1.82 | 50.43±1.07 | 41.63±1.74 | 43.68±0.68 | 41.35±0.65 | 74.87±0.25 | 60.67±1.64 |
| GAT | 49.11±2.93 | 34.67±3.87 | 61.33±3.48 | 53.04±3.06 | 59.77±1.97 | 46.63±1.54 | 52.93±1.22 | 48.54±0.46 | 71.83±0.22 | 62.56±3.54 |
| GIN | 47.11±4.59 | 34.22±1.78 | 50.43±1.92 | 35.59±3.75 | 46.23±0.88 | 37.29±1.04 | 43.91±0.45 | 41.24±0.52 | 48.17±0.44 | 41.00±2.75 |
| GraphSAGE | 47.33±3.03 | 39.33±3.11 | 70.96±1.48 | 60.35±1.83 | 58.91±1.92 | 49.23±1.90 | 51.92±0.26 | 47.44±0.83 | 71.24±3.00 | 56.63±5.81 |
| TopKPool | 44.22±2.76 | 36.00±4.80 | 55.25±2.74 | 43.83±5.21 | 44.83±2.19 | 34.63±2.08 | 48.97±1.24 | 42.87±1.31 | 56.69±3.58 | 33.49±1.74 |
| SAGPool | 46.67±2.53 | 37.11±5.00 | 55.71±4.71 | 39.30±5.49 | 41.89±4.28 | 30.17±1.85 | 50.01±0.68 | 45.16±0.36 | 59.91±4.14 | 24.62±0.32 |
| EdgePool | 51.11±3.06 | 33.33±1.99 | 64.17±2.44 | 55.36±2.16 | 56.74±3.98 | 45.89±1.30 | 50.17±0.64 | 45.90±0.44 | 76.96±0.13 | 62.31±1.23 |
| ASAP | 44.44±3.06 | 31.56±3.34 | 65.04±1.22 | 52.75±4.41 | 46.20±4.08 | 34.94±3.02 | 50.10±0.63 | 44.81±1.57 | 70.11±0.54 | 62.47±0.98 |
| Graph Transplant | 51.78±2.39 | 43.78±3.41 | 74.84±1.44 | 66.78±1.86 | 66.57±1.60 | 57.11±1.03 | 53.79±1.11 | 48.95±1.47 | 74.66±1.30 | 62.72±2.37 |
| PiCO | 46.88±2.76 | 35.78±3.02 | 73.56±1.71 | 64.63±4.35 | 76.25±1.66 | 63.69±1.42 | 53.47±1.14 | 46.04±1.20 | 53.99±0.92 | 34.74±1.77 |
| TGNN | 53.33±3.51 | 42.22±4.39 | 70.43±0.97 | 59.83±1.32 | 62.28±1.05 | 50.20±1.17 | OOM | OOM | 75.84±1.81 | 63.95±2.18 |
| GraphCL | 54.22±5.14 | 39.78±5.09 | 72.00±2.01 | 62.49±2.00 | 69.94±2.31 | 60.17±2.96 | 53.57±0.87 | 48.10±0.61 | 72.89±1.24 | 61.55±1.01 |
| GraphACL | 54.44±2.33 | 44.89±4.75 | 69.80±1.01 | 57.68±2.85 | 71.40±0.95 | 60.29±3.04 | 53.25±0.62 | 47.86±0.65 | 73.84±2.12 | 63.57±1.39 |
| DEER | 58.22±2.18 | 47.56±2.57 | 80.12±1.26 | 72.29±1.54 | 79.94±1.20 | 68.03±1.11 | 57.08±0.57 | 52.48±0.72 | 88.13±2.08 | 66.81±2.82 |
| **PRISM (Ours)** | **63.11**±0.83 | **51.33**±3.61 | **82.55**±0.89 | **78.32**±1.37 | **85.69**±1.06 | **79.48**±1.45 | **58.29**±0.83 | **55.10**±0.98 | **91.93**±2.26 | **80.57**±2.69 |

# 4 EXPERIMENTS

## 4.1 EXPERIMENTAL SETUP

**Datasets.** To comprehensively assess the performance of PRISM, we conduct experiments on five established graph classification benchmarks spanning the bioinformatics and vision domains: ENZYMES Schomburg et al. (2004), Letter-High Riesen & Bunke (2008), COIL-DEL Riesen & Bunke (2008), CIFAR10 Dwivedi et al. (2020), and COLORS-3 Knyazev et al. (2019). Following Gu et al. (2024), we randomly inject false positive labels into the candidate set to generate partial-label data. Specifically, each incorrect label $\overline{y} \neq y$ is included with probability $q = P(\overline{y} \in \mathcal{Y} \mid \overline{y} \neq y)$, controlling the degree of label ambiguity. A higher $q$ indicates noisier candidate supervision. We set $q \in \{0.1, 0.3, 0.5\}$ for most datasets, and $q \in \{0.02, 0.05, 0.1\}$ for COIL-DEL to account for its larger label space. More details about the datasets are provided in the Appendix E.

**Baseline Methods.** We compare our proposed PRISM with a comprehensive set of baselines across multiple paradigms: (a) *Graph neural networks*: GCN Welling & Kipf (2016), GAT Veličković et al. (2017), GIN Xu et al. (2018), and GraphSAGE Hamilton et al. (2017); (b) *Hierarchical graph pooling methods*: TopKPool Gao & Ji (2019), SAGPool Lee et al. (2019), EdgePool Diehl (2019), and ASAP Ranjan et al. (2020), all using GraphSAGE as the backbone; (c) *Graph augmentation method*: Graph Transplant Park et al. (2022); (d) *Unsupervised contrastive graph learning*: GraphCL You et al. (2020) and GraphACL Luo et al. (2023); (e) *Semi-supervised graph learning*: TGNN Ju et al. (2023); (f) *Partial label learning in vision*: PiCO Wang et al. (2022), adapted to the graph setting with a GraphSAGE encoder for fair comparison; (g) *Partial label learning for graphs*: DEER Gu et al. (2024). Further details on the baselines are provided in the Appendix F.

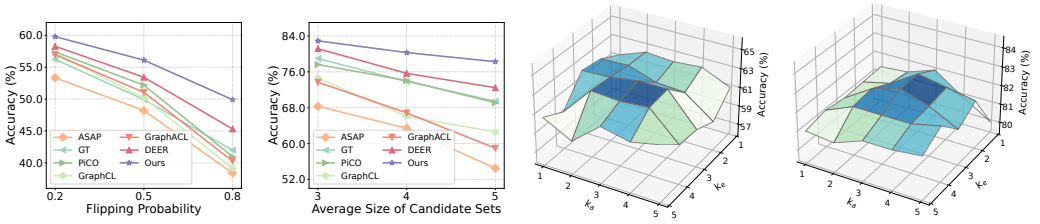
**Implementation and Evaluation Protocol.** All experiments are implemented using PyTorch with the PyG backend. We use a two-layer GraphSAGE with 512 hidden units as the shared encoder across all models. Training is performed using the Adam optimizer Kingma & Ba (2014) with an initial learning rate of 0.001 and a batch size of 128. All reported results are averaged over five independent runs with different random seeds, each reporting mean accuracy and standard deviation. More details about the implementation are provided in the Appendix G.

## 4.2 PERFORMANCE COMPARISON

Table 1 summarizes the classification performance of PRISM against diverse baselines under varying label ambiguity levels ($q$). The results highlight four key findings: (1) PRISM achieves consistently superior performance across all datasets and ambiguity settings, surpassing baselines by a considerable margin. For example, on ENZYMES, our method outperforms the second-best by

Table 2: Ablation study on ENZYMES, Letter-High, and CIFAR10.

| Datasets | ENZYMES | | Letter-High | | CIFAR10 | |
|---|---|---|---|---|---|---|
| Variants | $q = 0.3$ | $q = 0.5$ | $q = 0.3$ | $q = 0.5$ | $q = 0.3$ | $q = 0.5$ |
| PRISM w/o Sub | 61.78$\pm$2.06 | 49.56$\pm$3.55 | 80.70$\pm$1.20 | 76.11$\pm$2.19 | 56.65$\pm$1.01 | 53.30$\pm$0.77 |
| PRISM w/o Spa | 60.89$\pm$2.26 | 48.00$\pm$3.87 | 79.65$\pm$1.21 | 74.72$\pm$0.99 | 55.23$\pm$1.19 | 51.72$\pm$1.08 |
| PRISM w/o Spe | 61.55$\pm$1.94 | 48.89$\pm$2.11 | 80.17$\pm$0.43 | 76.46$\pm$2.45 | 55.72$\pm$0.93 | 52.65$\pm$1.12 |
| PRISM w/o Rel. Infer | 57.78$\pm$2.81 | 45.11$\pm$3.95 | 78.43$\pm$1.03 | 71.24$\pm$2.18 | 53.61$\pm$1.32 | 49.79$\pm$1.27 |
| **PRISM (Full Model)** | **63.11**$\pm$0.83 | **51.33**$\pm$3.61 | **82.55**$\pm$0.89 | **78.32**$\pm$1.37 | **58.29**$\pm$0.83 | **55.10**$\pm$0.98 |




(a) CIFAR10 (b) Letter-High (c) ENZYMES (d) Letter-High

Figure 2: (a) Performance comparison in scenarios with hierarchical label noise. (b) Performance comparison in scenarios with competitive label noise. (c)(d) Performance *w.r.t.* top-$k_a$ and top-$k_e$ on ENZYMES and Letter-High.

8.3% under $q = 0.3$, confirming its robustness and generalization under weak supervision. (2) On fine-grained datasets such as COIL-DEL, where class granularity and semantic ambiguity are pronounced, PRISM retains a strong advantage even at high noise. On COIL-DEL with $q = 0.1$, it attains 79.48% accuracy, exceeding the prior best (DEER, 68.03%) by 16.8%, showing effective candidate disambiguation via complementary spectral and substructure cues. (3) Compared to vision-based partial-label methods (e.g., PiCO) and semi-supervised models (e.g., TGNN), our framework consistently improves performance across all graph datasets. Although PiCO with GraphSAGE encoders is adapted to graphs, it still lags, underscoring the limits of directly transferring CV-based methods and the necessity of graph-specific modeling. (4) As label ambiguity $q$ increases, most methods degrade sharply, whereas PRISM exhibits a much slower decline. This robustness stems from integrating spectral and substructural reasoning, with confidence-aware propagation mitigating noisy and misleading supervision. More results under $q = 0.1$ are provided in Appendix D.

**Semantically Correlated Label Noise.** In many real-world scenarios, labels are not independent but semantically correlated, leading to candidate sets that contain noisy labels with stronger affinity to the ground-truth than unrelated classes, thereby creating additional challenges for robust learning. This raises the challenge of whether PRISM can reliably address such semantically entangled ambiguity. To evaluate this, we design two experimental protocols: (1) *Hierarchical label noise.* We exploit the coarse-to-fine taxonomy of CIFAR10 (e.g., vehicles vs. animals) and introduce noise by flipping negative labels within the same super-class as the true label with probability $q$, thereby forming semantically plausible candidates. (2) *Competitive label noise.* Following Yan & Guo (2023), we pretrain a graph neural network (GNN) on clean data to capture inter-class semantic dependencies. We then randomly select noisy candidates from the Top-$K$ predictions of this GNN, with $K = 6$ for Letter-High, and vary the candidate set size by adjusting sampling ratios. The results, presented in Figure 2 (a)(b), show that PRISM consistently outperforms all baselines under both noise schemes across different ambiguity levels, thereby highlighting its robustness to semantically correlated label noise. Details of the CIFAR10 hierarchy are provided in Appendix E.

### 4.3 Ablation Study

To assess the contribution of each core component in PRISM, we conduct ablation experiments on ENZYMES, Letter-High, and CIFAR10 under noise levels $q = 0.3$ and $q = 0.5$, with results in Table 2. Removing substructure alignment (PRISM w/o Sub) and replacing class-specific matching with holistic graph embeddings causes clear performance drops, especially under high ambiguity, showing that discriminative substructure cues outperform coarse comparisons. Excluding the spatial branch (PRISM w/o Spa) further reduces accuracy, as the absence of neighborhood-aware

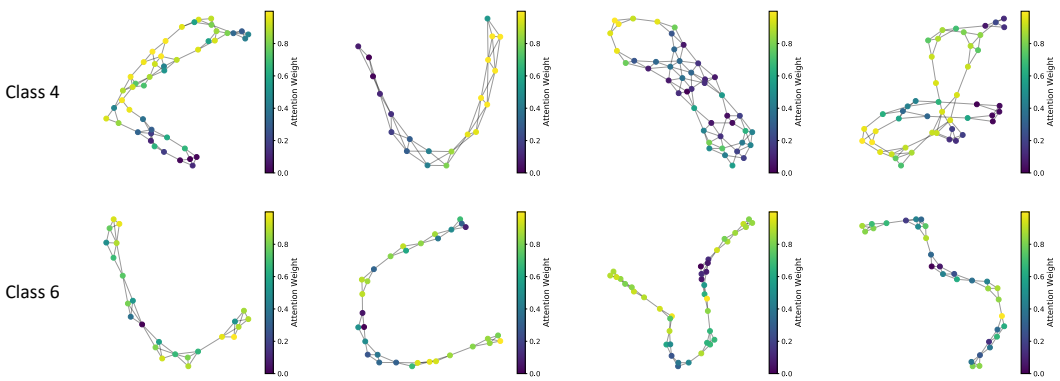



Figure 3: Class-specific attention maps reveal consistent focus on discriminative substructures across graph instances within different categories.

signals weakens the capture of fine-grained topology critical for resolving label ambiguity, particularly in CIFAR10. Discarding the spectral attention module (PRISM w/o Spe) validates the value of frequency-specific reasoning, since multi-band decomposition highlights global semantics complementing local structure. Most critically, removing the relational inference layer (PRISM w/o Rel. Infer) yields the sharpest degradation, especially at $q = 0.5$, underscoring that graph-of-graph modeling and label propagation are indispensable for suppressing noise and ensuring robust supervision.

### 4.4 Sensitivity Analysis

To investigate the influence of relational sparsity, we analyze how the number of neighbors retained in the spatial ($k_a$) and spectral ($k_e$) relational graphs affects performance. As illustrated in Figure 2 (c)(d), accuracy generally increases as $k_a$ and $k_e$ grow from 1 to moderate values, then saturates or slightly declines. This pattern indicates that adding structurally or spectrally aligned instances improves cross-graph consistency by enhancing the density of reliable signals. However, overly dense connectivity introduces noisy or weakly correlated neighbors, diluting discriminative patterns and potentially causing label propagation drift. Our model achieves stable performance under a broad range of $k$ values, reflecting robustness to hyperparameter choices in multi-view graph construction.

### 4.5 Visualization of Class-Specific Attention

To assess the interpretability of our spatial encoder, we visualize class-conditioned attention maps from the ENZYMES dataset. As shown in Figure 3, each node is colored by its attention weight relative to the ground-truth class prototype, with warmer colors denoting greater contribution to class-specific substructure reasoning. The model consistently highlights structurally informative regions, such as densely connected motifs, central connectors, or bridging nodes, while down-weighting peripheral or less relevant parts of the graph. Despite structural variability across samples, attention patterns remain highly consistent within the same class, indicating that prototype-guided attention effectively identifies semantically aligned substructures relevant for enzyme classification and provides human-interpretable insights into functionally critical topological regions.

## 5 Conclusion

In this work, we study the underexplored problem of partial-label graph learning, where each graph is annotated with an ambiguous candidate label set. To address the inherent challenges of semantic uncertainty and structural complexity, we propose PRISM, a unified relational inference framework that integrates spatial substructure alignment and spectral frequency modeling via dual relational graphs. Our method leverages confidence-aware label propagation and candidate-constrained refinement to disambiguate supervision without relying on ground-truth annotations or external auxiliary signals. Extensive experiments across multiple benchmarks validate the effectiveness, robustness, and generalizability of PRISM under various ambiguity levels, highlighting its potential as a principled and versatile solution for learning from weakly supervised graph data.

## Ethics Statement

We acknowledge that all co-authors of this work have read and committed to adhering to the ICLR Code of Ethics.

## Reproducibility Statement

We have included all details about the datasets and our experiment settings in Appendix E and Appendix G, respectively. The anonymous source code can be found in https://anonymous.4open.science/r/PRISM-17107/. The code and related materials will be appropriately released to ensure the transparency and reproducibility of our work.

## References


Axel D Becke. Perspective: Fifty years of density-functional theory in chemical physics. *The Journal of chemical physics*, 140(18):18A301, 2014.

Deyu Bo, Yuan Fang, Yang Liu, and Chuan Shi. Graph contrastive learning with stable and scalable spectral encoding. *Advances in Neural Information Processing Systems*, 36:45516–45532, 2023.

Dapeng Chen, Min Wang, Haobin Chen, Lin Wu, Jing Qin, and Wei Peng. Cross-modal retrieval with heterogeneous graph embedding. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 3291–3300, 2022.

Timothee Cour, Ben Sapp, and Ben Taskar. Learning from partial labels. *The Journal of Machine Learning Research*, 12:1501–1536, 2011.

Frederik Diehl. Edge contraction pooling for graph neural networks. *arXiv preprint arXiv:1905.10990*, 2019.

Zilin Du, Yunxin Li, Xu Guo, Yidan Sun, and Boyang Li. Training multimedia event extraction with generated images and captions. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 5504–5513, 2023.

Vijay Prakash Dwivedi, Chaitanya K Joshi, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *arXiv preprint arXiv:2003.00982*, 2020.

Xiaomin Fang, Lihang Liu, Jieqiong Lei, Donglong He, Shanzhuo Zhang, Jingbo Zhou, Fan Wang, Hua Wu, and Haifeng Wang. Geometry-enhanced molecular representation learning for property prediction. *Nature Machine Intelligence*, 4(2):127–134, 2022.

Lei Feng and Bo An. Partial label learning by semantic difference maximization. In *IJCAI*, pp. 2294–2300, 2019.

Hang Gao, Jiaguo Yuan, Jiangmeng Li, Peng Qiao, Fengge Wu, Changwen Zheng, and Huaping Liu. Graph partial label learning with potential cause discovering. *arXiv preprint arXiv:2403.11449*, 2024.

Hongyang Gao and Shuiwang Ji. Graph u-nets. In *international conference on machine learning*, pp. 2083–2092. PMLR, 2019.

Lingchi Ge, Min Fang, Haikun Li, and Bo Chen. Label correlation for partial label learning. *Journal of Systems Engineering and Electronics*, 33(5):1043–1051, 2022.

Yiyang Gu, Zihao Chen, Yifang Qin, Zhengyang Mao, Zhiping Xiao, Wei Ju, Chong Chen, Xian-Sheng Hua, Yifan Wang, Xiao Luo, et al. Deer: Distribution divergence-based graph contrast for partial label learning on graphs. *IEEE Transactions on Multimedia*, 2024.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

Eyke Hüllermeier and Jürgen Beringer. Learning from ambiguously labeled examples. In *Advances in Intelligent Data Analysis VI: 6th International Symposium on Intelligent Data Analysis, IDA 2005, Madrid, Spain, September 8-10, 2005. Proceedings 6*, pp. 168–179. Springer, 2005.

Wei Ju, Xiao Luo, Meng Qu, Yifan Wang, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. Tgnn: A joint semi-supervised framework for graph-level classification. *arXiv preprint arXiv:2304.11688*, 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Boris Knyazev, Graham W Taylor, and Mohamed Amer. Understanding attention and generalization in graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Dongha Lee, Su Kim, Seonghyeon Lee, Chanyoung Park, and Hwanjo Yu. Learnable structural semantic readout for graph classification. In *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 1180–1185. IEEE, 2021.

Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *International conference on machine learning*, pp. 3734–3743. PMLR, 2019.

Sihang Li, Xiang Wang, An Zhang, Yingxin Wu, Xiangnan He, and Tat-Seng Chua. Let invariant rationale discovery inspire graph contrastive learning. In *International conference on machine learning*, pp. 13052–13065. PMLR, 2022.

Meng Liu, Ke Liang, Dayu Hu, Hao Yu, Yue Liu, Lingyuan Meng, Wenxuan Tu, Sihang Zhou, and Xinwang Liu. Tmac: Temporal multi-modal graph learning for acoustic event classification. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 3365–3374, 2023.

Xiao Luo, Wei Ju, Yiyang Gu, Zhengyang Mao, Luchen Liu, Yuhui Yuan, and Ming Zhang. Self-supervised graph-level representation learning with adversarial contrastive learning. *ACM Transactions on Knowledge Discovery from Data*, 18(2):1–23, 2023.

Gengyu Lyu, Songhe Feng, Tao Wang, and Congyan Lang. A self-paced regularization framework for partial-label learning. *IEEE Transactions on Cybernetics*, 52(2):899–911, 2020.

Mourad Oudghiri. Weyl's theorem and perturbations. *Integral Equations and Operator Theory*, 53 (4):535–545, 2005.

Joonhyung Park, Hajin Shim, and Eunho Yang. Graph transplant: Node saliency-guided graph mixup with local structure preservation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7966–7974, 2022.

Ekagra Ranjan, Soumya Sanyal, and Partha Talukdar. Asap: Adaptive structure aware pooling for learning hierarchical graph representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5470–5477, 2020.

Kaspar Riesen and Horst Bunke. Iam graph database repository for graph based pattern recognition and machine learning. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pp. 287–297. Springer, 2008.

François Rousseau, Emmanouil Kiagias, and Michalis Vazirgiannis. Text categorization as a graph classification problem. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1702–1712, 2015.

Ida Schomburg, Antje Chang, Christian Ebeling, Marion Gremse, Christian Heldt, Gregor Huhn, and Dietmar Schomburg. Brenda, the enzyme database: updates and major new developments. *Nucleic acids research*, 32(suppl_1):D431–D433, 2004.

Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(9), 2011.

Vinesh Solanki, Patrick Rubin-Delanchy, and Ian Gallagher. Persistent homology of graph embeddings, 2021. URL https://arxiv.org/abs/1912.10238.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, and Junbo Zhao. Pico: Contrastive label disambiguation for partial label learning. In *Proceedings of the International Conference on Learning Representations*, 2022.

Qifan Wang, Yinwei Wei, Jianhua Yin, Jianlong Wu, Xuemeng Song, and Liqiang Nie. Dualgnn: Dual graph neural network for multimedia recommendation. *IEEE Transactions on Multimedia*, 25:1074–1084, 2021.

Wei Wang and Min-Ling Zhang. Partial label learning with discrimination augmentation. In *Proceedings of the International ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1920–1928, 2022.

Max Welling and Thomas N Kipf. Semi-supervised classification with graph convolutional networks. In *J. International Conference on Learning Representations (ICLR 2017)*, 2016.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2018.

Yan Yan and Yuhong Guo. Mutual partial label learning with competitive label noise. In *The Eleventh International Conference on Learning Representations*, 2023.

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

Guoxian Yu, Hailong Zhu, and Carlotta Domeniconi. Predicting protein functions using incomplete hierarchical labels. *BMC bioinformatics*, 16(1):1–12, 2015a.

Yi Yu, Tengyao Wang, and Richard J Samworth. A useful variant of the davis–kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015b.

Tong Zhang, Yun Wang, Zhen Cui, Chuanwei Zhou, Baoliang Cui, Haikuan Huang, and Jian Yang. Deep wasserstein graph discriminant learning for graph classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10914–10922, 2021a.

Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Chee-Kong Lee. Motif-based graph self-supervised learning for molecular property prediction. *Advances in Neural Information Processing Systems*, 34:15870–15882, 2021b.

## A Proof of Theorem 1

To begin with, we denote adjacent matrix of graph $i$ as $A^{(i)}$ for the sake of convenience, take READOUT function as the mean in (2) that $g_i = \frac{1}{|\mathcal{V}_i|}\sum_{v\in\mathcal{V}_i} h_v^{(i)}$. Since $p_c$ is the prototype of label $c$, we assume the latent representations of nodes satisfy

$$\mathbb{E}\left[h_v^{(i)}|y_i^* = c\right] = p_c,\ \forall v \in \mathcal{V}_i. \tag{20}$$

where $y_i^*$ is the ground-truth label of graph $i$. Assume the latent representations of nodes in one graph are identically distributed, it follows from (20) and the law of large number that for any graph $i$ with true label $y_i^* = c$

$$g_i = \frac{1}{|\mathcal{V}_i|}\sum_{v\in\mathcal{V}_i} h_v^{(i)} \xrightarrow{a.s} \mathbb{E}\left[h_v^{(i)}|y_i^* = c\right] = p_c$$

as $|\mathcal{V}_i| \to \infty$, which is account for (3) to a certain degree. As for the adjacent matrix, we refer to Gu et al. (2024) to consider the generalized random dot product graph, which can date back to Solanki et al. (2021) that for graph $i$:

$$\mathbb{P}\left(A_{jk}^{(i)} = 1|\Theta\right) = \alpha_n \xi_j^T I_{r,-r}\xi_k$$

where $\Theta = (\xi_1, ..., \xi_n) \sim F^{(i)}, I_{r,-r} = \text{diag}\,(I_r, -I_r)$. We assume the distribution $F^{(i)}$ is determined by the global graph feature $g_i$ that there exist a continuous matrix function $\mathcal{M}$ such that

$$A^{(i)} = \mathcal{M}\left(g_i\right),\ \forall i \in \{1, 2, ..., N\}. \tag{21}$$

Take coefficients $\alpha_{vc}^{(i)} = \frac{1}{|\mathcal{V}_i|}$ and denote $\lambda_p^{(i)}, u_p^{(i)}$ as the $p$-th eigenvalue, eigenvector for graph $i$ for the sake of simplicity. First we consider if $y_i^* = c, y_j^* = c'$ then by (20) and the continuity theorem

$$\begin{aligned}\cos\left(r_i^{(c)}, r_j^{(c')}\right) &= \sum_{v\in\mathcal{V}_i, v'\in\mathcal{V}_j} \alpha_{vc}^{(i)}\alpha_{v'c'}^{(j)} \cos\left(h_v^{(i)}, h_{v'}^{(j)}\right) \\ &= \cos\left(\sum_{v\in\mathcal{V}_i}\frac{1}{|\mathcal{V}_i|}h_v^{(i)}, \frac{1}{|\mathcal{V}_j|}\sum_{v'\in\mathcal{V}_j} h_{v'}^{(j)}\right) \xrightarrow{a.s.} \cos\left(p_c, p_{c'}\right)\end{aligned}$$

as $|\mathcal{V}_i|, |\mathcal{V}_j| \to \infty$, and

$$\cos\left(r_i^{(c)}, p_c\right) = \sum_{v\in\mathcal{V}_i}\alpha_{vc}^{(i)}\cos\left(h_v^{(i)}, p_c\right) = \cos\left(\frac{1}{|\mathcal{V}_i|}\sum_{v\in\mathcal{V}_i} h_v^{(i)}, p_c\right) \xrightarrow{a.s.} \cos\left(p_c, p_c\right) = 1,$$

as $|\mathcal{V}_i| \to \infty$, which further implies

$$\begin{aligned}&\mathbb{P}\left(A_{ij}^{spa} = 1|y_i^* = y_j^*\right) = \sum_{c=1}^{C}\mathbb{P}\left(A_{ij}^{spa} = 1|y_i^* = y_j^* = c\right)\mathbb{P}\left(y_i^* = y_j^* = c|y_i^* = y_j^*\right) \\ &\geq \sum_{c=1}^{C}\mathbb{P}\left(s_{ij}^{spa} = 1|y_i^* = y_j^* = c\right)\mathbb{P}\left(y_i^* = y_j^* = c|y_i^* = y_j^*\right) \\ &\geq \sum_{c=1}^{C}\mathbb{P}\left(\cos\left(r_i^{(c)}, r_j^{(c)}\right) = \cos\left(\frac{r_i^{(c)} + r_j^{(c)}}{2}, p_c\right) = 1|y_i^* = y_j^* = c\right)\mathbb{P}\left(y_i^* = y_j^* = c|y_i^* = y_j^*\right) \\ &\to \sum_{c=1}^{C}\mathbb{P}\left(y_i^* = y_j^* = c|y_i^* = y_j^*\right) = 1,\end{aligned}$$

as $|\mathcal{V}_i|, |\mathcal{V}_j| \to \infty$, where the second and third line follow from the definitions of $A^{spa}$ and $s_{ij}^{spa}$. As for (12), observe that (21) indicates the adjacent matrix of graph $i$ given $y_i^* = c$ satisfies

$$A^{(i)} = \mathcal{M}\left(\frac{1}{|\mathcal{V}_i|}\sum_{v\in\mathcal{V}_i} h_v^{(i)}\right) \xrightarrow{a.s.} \mathcal{M}\left(p_c\right)$$

as $|\mathcal{V}_i| \to \infty$, which implies $||A^{(i)} - A^{(j)}||_F \xrightarrow{a.s.} 0$ given $y_i^* = y_j^* = c$. Further by Weyl's Perturbation Theorem in Oudghiri (2005) we have

$$\left|\lambda_p^{(i)} - \lambda_p^{(j)}\right| \leq ||A^{(i)} - A^{(j)}||_F \xrightarrow{a.s.} 0,$$

while by a variant of Davis-Kahan Theorem, Theorem 2 in Yu et al. (2015b) and $\min_{1\leq i\leq N, 1\leq p\leq T}\left|\lambda_{p+1}^{(i)} - \lambda_p^{(i)}\right| \geq \delta$ we have

$$||u_p^{(i)} - u_p^{(j)}|| \leq \frac{4}{\delta}||A^{(i)} - A^{(j)}||_F \xrightarrow{a.s.} 0$$

for $1 \leq p \leq T$. Therefore by (8) and the continuity theorem we have

$$||z_i^{(p)} - z_j^{(p)}|| \xrightarrow{a.s.} 0$$

for $1 \leq p \leq T$, which leads to

$$1 \geq \cos\left(z_i^{(p)}, z_j^{(p)}\right) = \frac{||z_i^{(p)}||^2 + ||z_j^{(p)}||^2 - ||z_i^{(p)} - z_j^{(p)}||^2}{2||z_i^{(p)}|| \cdot ||z_j^{(p)}||} \xrightarrow{a.s.} \frac{||z_i^{(p)}||^2 + ||z_j^{(p)}||^2}{2||z_i^{(p)}|| \cdot ||z_j^{(p)}||} \geq 1.$$

Then by the definitions of $A^{spe}$ and $s_{ij}^{spe}$ we obtain

$$\begin{aligned}
&\mathbb{P}\left(A_{ij}^{spe} = 1 | y_i^* = y_j^*\right) = \sum_{c=1}^{C} \mathbb{P}\left(A_{ij}^{spe} = 1 | y_i^* = y_j^* = c\right) \mathbb{P}\left(y_i^* = y_j^* = c | y_i^* = y_j^*\right) \\
&\geq \sum_{c=1}^{C} \mathbb{P}\left(s_{ij}^{spe} = 1 | y_i^* = y_j^* = c\right) \mathbb{P}\left(y_i^* = y_j^* = c | y_i^* = y_j^*\right) \\
&\geq \sum_{c=1}^{C} \mathbb{P}\left(\cos\left(z_i^{(p)}, z_j^{(p)}\right) = 1, p = 1, ..., T | y_i^* = y_j^* = c\right) \mathbb{P}\left(y_i^* = y_j^* = c | y_i^* = y_j^*\right) \\
&\to \sum_{c=1}^{C} \mathbb{P}\left(y_i^* = y_j^* = c | y_i^* = y_j^*\right) = 1,
\end{aligned}$$

as $|\mathcal{V}_i|, |\mathcal{V}_j| \to \infty$, which completes the proof. □

## B Proof of Theorem 2

Recall that we assume the classifier is well-trained such that

$$f_{classifier}\left(p_c\right) = \mathbb{I}_c, \ \forall c \in \mathcal{Y}, \tag{22}$$

where $\mathbb{I}_c \in \{0,1\}^C$ denotes one-hot vector whose $c$-th component is 1 and the rest are 0. By Theorem 1 and (13) we have

$$Y_i^{(T)} \xrightarrow{a.s.} Y_i^*$$

as $|\mathcal{V}_i|, T \to \infty$, which further implies $Q_i \xrightarrow{a.s.} Y_i^*$ according to (15). By the continuity theorem and (22) we have

$$P_i = f_{classifier}\left(g_i\right) = f_{classifier}\left(\sum_{v\in\mathcal{V}_i} \frac{1}{|\mathcal{V}_i|} h_v^{(i)}\right) \to f_{classifier}\left(p_{y_i^*}\right) = \mathbb{I}_{y_i^*} = Y_i^* \tag{23}$$

as $|\mathcal{V}_i| \to \infty$. Denote $|\mathcal{V}| = \min_{1\leq i\leq N} |\mathcal{V}_i|$, by dominated convergence theorem and (16) we obtain

$$\begin{aligned}
\lim_{|\mathcal{V}|,T\to\infty} \mathbb{E}\left[\mathcal{L}_{sup}\right] &= -\frac{1}{B}\sum_{i=1}^{B} \lim_{|\mathcal{V}|,T\to\infty} \mathbb{E}\left[\log \sum_{c\in S_i} \text{Softmax}\left(P_i\right)_c \cdot Q_{ic}\right] \\
&= -\frac{1}{B}\sum_{i=1}^{B} \mathbb{E}\left[\lim_{|\mathcal{V}|,T\to\infty} \log\left(\sum_{c\neq y_i^*} \text{Softmax}\left(P_i\right)_c \cdot Q_{ic} + \left(P_i\right)_{y_i^*} \cdot Q_{i,y_i^*}\right)\right] \\
&= -\frac{1}{B}\sum_{i=1}^{B} \mathbb{E} \log\left[0 + 1\right] = 0
\end{aligned}$$

where the last line follows from (18) and (23). □

# C Related Work

## C.1 Graph Classification

Graph classification has been widely applied in fields such as molecular property prediction, protein interaction analysis, and social network modeling Fang et al. (2022); Wang et al. (2021). Traditional graph kernel methods Shervashidze et al. (2011) measure structural similarity through subgraph comparisons, but scale poorly to large graphs. Recent advances in Graph Neural Networks (GNNs) Welling & Kipf (2016); Hamilton et al. (2017); Veličković et al. (2017); Xu et al. (2018) have shown superior performance by aggregating local neighborhood information and generating graph-level representations via pooling operators Lee et al. (2019); Gao & Ji (2019); Lee et al. (2021). Spectral approaches like EigenMLP Bo et al. (2023) provide an alternative by encoding global graph structure through Fourier-like eigenvalue embeddings. Despite their success, these methods heavily rely on clean and abundant labels, which are often unavailable in real-world scenarios due to annotation cost or inherent uncertainty. While self-supervised methods such as GraphCL You et al. (2020) avoid labels during pre-training, they still require accurate annotations for downstream classification and tend to degrade significantly under label ambiguity. Our work takes a step further by addressing the graph classification task under partial-label settings with relational inference over spatial and spectral cues.

## C.2 Partial Label Learning

Partial label learning (PLL) considers a weak supervision setting where each training instance is annotated with a candidate label set containing only one correct label Hüllermeier & Beringer (2005); Cour et al. (2011). Early approaches treat all candidates equally by averaging losses, but such uniform assumptions often fail under high label ambiguity. Later works focus on disambiguation, estimating true labels through confidence-based or similarity-driven refinement Feng & An (2019); Wang & Zhang (2022). Recent advances introduce contrastive learning to PLL Wang et al. (2022), where prototype-instance alignment helps separate correct labels from distractors. However, these methods are mostly designed for Euclidean data such as images or texts. On graph-structured data, PLL remains underexplored. DEER Gu et al. (2024) is one of the few attempts, proposing to measure semantic distribution divergence between graph views for contrastive learning and using posterior-guided soft label correction. Nonetheless, DEER relies on semantic distribution matching and lacks fine-grained structural modeling. Another work, GPCD Gao et al. (2024), introduces graph potential cause discovery to estimate causal subsets for supervision, but suffers from high training complexity and overlooks global spectral cues. In contrast, PRISM integrates spatial substructure information and spectral semantics into a unified relational inference framework, enabling more precise and robust label disambiguation under complex graph structures.

# D More Experimental Results

We present additional results under the low ambiguity level ($q = 0.1$, $q = 0.02$) in Table 3. Across all datasets, PRISM consistently achieves the best performance, reaffirming its robustness under mild supervision noise. In particular, clear gains are observed on fine-grained datasets such as COIL-DEL, where both spatial and spectral cues play a pivotal role in resolving semantic ambiguities. These results complement the main findings in Section 4.2, further demonstrating that our framework sustains strong generalization across diverse noise regimes.

# E Details of Datasets

To rigorously evaluate the effectiveness of our proposed PRISM, we conduct extensive experiments on four graph classification benchmarks spanning bioinformatics and visual domains: ENZYMES, Letter-High, COIL-DEL, and COLORS-3. These datasets provide a wide range of structural configurations and semantic granularities, enabling a comprehensive analysis of PRISM under varied supervision conditions.

- **ENZYMES** Schomburg et al. (2004) is a bioinformatics dataset comprising 600 protein tertiary structures. Each graph represents a protein, where nodes correspond to secondary structure ele-

Table 3: The classification accuracy (mean%±std%) on five graph benchmark datasets. The best results are shown in boldface and the second best results are underlined. $q = P(\overline{y} \in Y | \overline{y} \neq y)$ reflecting the degree of label ambiguity.

| Dataset | ENZYMES | Letter-High | COIL-DEL | CIFAR10 | COLORS-3 |
|---|---|---|---|---|---|
| Methods | $q = 0.1$ | $q = 0.1$ | $q = 0.02$ | $q = 0.1$ | $q = 0.1$ |
| GCN | 61.33±2.85 | 50.09±0.70 | 60.77±1.71 | 47.18±1.09 | 90.34±0.06 |
| GAT | 58.22±3.03 | 73.39±1.41 | 69.11±2.86 | 57.56±0.65 | 89.33±1.42 |
| GIN | 59.78±4.58 | 55.83±4.28 | 55.94±1.69 | 47.29±0.61 | 63.01±1.45 |
| GraphSAGE | 60.89±1.09 | 78.20±1.17 | 71.40±2.15 | 57.22±0.67 | 91.70±2.18 |
| TopKPool | 53.11±4.12 | 67.07±1.60 | 55.80±4.86 | 55.26±0.85 | 82.35±1.36 |
| SAGPool | 56.89±5.37 | 67.42±1.91 | 52.94±2.59 | 54.23±0.53 | 76.99±4.39 |
| EdgePool | 58.67±2.67 | 70.49±3.29 | 68.74±1.85 | 55.09±0.61 | 87.47±0.41 |
| ASAP | 60.89±2.67 | 71.25±1.44 | 59.03±3.09 | 54.56±0.66 | 77.84±1.26 |
| Graph Transplant | 61.56±2.86 | 80.75±0.60 | 80.09±0.75 | 56.87±1.28 | 85.48±0.89 |
| PiCO | 61.08±6.67 | 81.27±1.60 | 84.88±1.09 | 57.70±0.82 | 65.68±1.07 |
| TGNN | 62.44±3.01 | 78.55±0.78 | 70.49±0.87 | OOM | 93.16±1.55 |
| GraphCL | 61.78±1.51 | 78.43±0.85 | 78.83±1.06 | 57.62±0.56 | 92.71±1.61 |
| GraphACL | 58.22±1.51 | 81.04±1.01 | 80.66±0.41 | 57.65±0.21 | 92.05±0.50 |
| DEER | 67.11±1.66 | 83.48±0.92 | 87.86±1.41 | 61.45±0.40 | 96.23±2.94 |
| **PRISM (Ours)** | **68.00**±1.63 | **84.87**±0.74 | **89.29**±1.36 | **61.73**±0.87 | **98.36**±1.07 |

ments (e.g., helices, strands), and edges indicate either spatial proximity or sequential adjacency. The classification involves assigning one of six enzyme commission (EC) classes.

- **Letter-High** Riesen & Bunke (2008) contains graphs constructed from 15 uppercase letters. Each letter is transformed into a prototype graph by representing stroke endpoints as nodes and line segments as edges. The dataset emphasizes shape topology and inter-class similarity among characters.
- **COIL-DEL** Riesen & Bunke (2008) is derived from object images by applying Harris corner detection followed by Delaunay triangulation. This results in undirected graphs where nodes denote detected corners and edges reflect geometric connectivity. The dataset includes 100 object categories with significant structural variation.
- **CIFAR10** Dwivedi et al. (2020) is constructed from an image classification benchmark by transforming each image into a graph representation based on superpixels. In this formulation, nodes denote individual superpixels, and edges are established according to $k$-nearest neighbor relationships. The dataset further exhibits a hierarchical taxonomy: the first super-class (*vehicles*) comprises airplane, automobile, ship, and truck (excluding pickup truck), whereas the second super-class (*animals*) encompasses bird, cat, deer, dog, frog, and horse.
- **COLORS-3** Knyazev et al. (2019) is a synthetic benchmark for evaluating reasoning over discrete node attributes. Each graph contains nodes with categorical color features (red, green, blue), encoded as one-hot vectors. The task is to count the number of nodes of a specified color, requiring models to identify and aggregate attribute-specific information.

To emulate the partial-label learning scenario, we adopt the controlled label corruption protocol in Gu et al. (2024), wherein each sample is provided with a candidate label set that includes the ground-truth label and randomly sampled distractors. The inclusion probability $q = P(\overline{y} \in \mathcal{S}_i \mid \overline{y} \neq y_i)$ determines the ambiguity level of the candidate supervision. We set $q \in \{0.1, 0.3, 0.5\}$ for ENZYMES, Letter-High, and COLORS-3, while smaller values $\{0.02, 0.05, 0.1\}$ are chosen for COIL-DEL due to its larger label space and increased visual complexity.

## F Details of Baselines

We compare our proposed PRISM against a broad spectrum of baseline models categorized into seven distinct groups: (a) *Graph neural networks*: GCN Welling & Kipf (2016), GAT Veličković et al. (2017), GIN Xu et al. (2018), and GraphSAGE Hamilton et al. (2017); (b) *Hierarchical graph pooling*: TopKPool Gao & Ji (2019), SAGPool Lee et al. (2019), EdgePool Diehl (2019), and

ASAP Ranjan et al. (2020) (all using GraphSAGE as the encoder backbone); (c) *Graph augmentation method*: Graph Transplant Park et al. (2022); (d) *Contrastive graph learning*: GraphCL You et al. (2020) and GraphACL Luo et al. (2023); (e) *Weakly-supervised learning*: TGNN Ju et al. (2023); (f) *Partial-label learning in vision*: PiCO Wang et al. (2022), adapted with a GraphSAGE encoder; (g) *Partial-label learning for graphs*: DEER Gu et al. (2024), which operates directly on the partial supervision scenario. These baselines span diverse learning paradigms and offer complementary modeling assumptions, providing a comprehensive testbed for evaluation.

- **GCN** Welling & Kipf (2016): A spectral convolutional network that leverages renormalized adjacency matrices to aggregate first-order neighborhood information in a computationally efficient manner.
- **GAT** Veličković et al. (2017): Introduces attention weights over neighbors, enabling each node to prioritize important neighbors during message passing.
- **GIN** Xu et al. (2018): Uses MLPs to approximate injective functions over multisets of neighbors, achieving powerful discriminative capacity aligned with the Weisfeiler-Lehman graph test.
- **GraphSAGE** Hamilton et al. (2017): Aggregates information from randomly sampled neighbors and generalizes to unseen graphs via inductive learning, making it scalable to large datasets.
- **TopKPool** Gao & Ji (2019): Selects top-ranked nodes based on a learnable projection score to form a pooled graph with reduced size and preserved discriminative regions.
- **SAGPool** Lee et al. (2019): Utilizes self-attention scores computed from graph convolutions to guide node selection for pooling, capturing both feature and structural signals.
- **EdgePool** Diehl (2019): Contracts informative edges iteratively to coarsen the graph while maintaining crucial topological structures.
- **ASAP** Ranjan et al. (2020): Combines node selection and clustering by learning soft assignments over local $h$-hop neighborhoods, allowing for adaptive structure-aware pooling.
- **Graph Transplant** Park et al. (2022): A mixup-inspired data augmentation strategy that extracts meaningful subgraphs based on node saliency and generates hybrid samples through substructure-level interpolation.
- **GraphCL** You et al. (2020): A contrastive learning framework that maximizes agreement between different augmented views of a graph, using stochastic transformations and InfoNCE loss.
- **GraphACL** Luo et al. (2023): Improves contrastive representation learning by constructing adversarial hard negatives and regularizing the feature space with orthogonality and divergence constraints.
- **TGNN** Ju et al. (2023): A dual-view semi-supervised framework that integrates message passing and kernel-based reasoning, encouraging consistency across views to exploit both labeled and unlabeled graphs.
- **PiCO** Wang et al. (2022): Learns a set of class-wise prototypes and employs contrastive objectives to align instance embeddings with their correct prototypes. We adopt a GraphSAGE encoder to enable graph-level application.
- **DEER** Gu et al. (2024): A partial-label graph learning method that selects reliable positive pairs by measuring distribution divergence across augmented views. It also performs soft label correction via posterior estimation. However, it does not explicitly model substructures or spectral signals, limiting its granularity in structural reasoning.

**Training Protocol.** All baselines are trained under partial-label supervision using a cross-entropy loss over candidate label sets. TGNN additionally optimizes a consistency loss between dual views. For contrastive methods such as GraphCL and PiCO, we apply their contrastive loss and the cross-entropy loss. Pooling-based methods are evaluated with a fixed reduction ratio of $0.6$ and use GraphSAGE as their encoder. All models are trained under identical random seeds for controlled comparisons.

## G Details of Implementation

We implement all models using PyTorch with the PyG backend. A two-layer GraphSAGE with 512 hidden units is adopted as the shared encoder. The model is trained using the Adam optimizer with a

learning rate of 0.001 and a batch size of 128. The EMA momentum for prototype and label updates is set to $m = 0.99$, and the propagation momentum is $\alpha = 0.9$ with $T = 2$ steps. Both spatial and spectral relational graphs connect each graph to $k_a = k_e = 5$ neighbors. Eigen-decomposition is precomputed via sparse solvers to ensure efficiency. For ENZYMES, Letter-High, COIL-DEL, and COLORS-3, we partition the data into training, validation, and test sets with a ratio of $80\%$:$5\%$:$15\%$. For CIFAR10, we adopt the conventional split of $45{,}000$ training, $5{,}000$ validation, and $10{,}000$ test graphs, consistent with the protocol in Dwivedi et al. (2020). All reported results are averaged over five independent random seeds.

## LLMs USAGE

We adhere to the ICLR Code of Ethics. We use large language models solely for polishing writing. All scientific contributions remain entirely our own.