Graph Contrastive Learning via Spectral Graph Alignment

Manh Nguyen
Department of Statistics
University of Wisconsin-Madison
mdnguyen4@wisc.edu

Abstract

Given augmented views of each input graph, contrastive learning methods (e.g., InfoNCE) optimize pairwise alignment of graph embeddings across views while providing no mechanism to control the global structure of the view specific graph-of-graphs built from these embeddings. We introduce **SpecMatch-CL**, a novel loss function that aligns the view specific graph-of-graphs by minimizing the difference between their normalized Laplacians. Theoretically, we show that under certain assumptions, the difference between normalized Laplacians provides an upper bound not only for the difference between the ideal Perfect Alignment contrastive loss and the current loss, but also for the Uniformly loss [1]. Empirically, SpecMatch-CL establishes new state of the art on eight TU benchmarks under unsupervised learning and semi-supervised learning at low label rates, and yields consistent gains in transfer learning on PPI-306K and ZINC 2M datasets. The Pytorch implementation for our method is provided in github.com/manhbeo/GNN-CL.

1 Introduction

Contrastive learning (CL) has become a dominant paradigm for representation learning across modalities, including graphs [2, 3, 4]. At its core, CL encourages the alignment of positive pairs while maintaining uniformity in the representation space [1]. In graphs, recent approaches realize these principles either by manipulating topology through augmentations (e.g. GraphCL and its automated variants JOAO/JOAOv2) or by contrasting local/global summaries and diffusion-based views [3, 5, 6, 7, 8]. However, the instance-level nature of InfoNCE leaves an important degree of freedom: two augmented views can attain similar contrastive objectives while inducing different global neighborhood structure (e.g., connectivity patterns, cluster margins, and multihop relations).

We address this gap by explicitly regularizing view-to-view spectral consistency. Our method, **SpecMatch-CL**, constructs sparse neighborhood graphs from the embeddings of each view and penalizes the spectral norm of the difference between their normalized Laplacians. Intuitively, if the spectra match, the induced diffusion geometries –and hence multiscale neighborhoods—are similar across views, removing degeneracies that alignment alone does not fix. To ground this design, we show that small Laplacian discrepancy implies not only a small discrepancy of the contrastive objective relative to its ideal Perfect Alignment counterpart but also a small Uniformity loss, thereby providing an optimization-agnostic rationale for the novel loss.

Contributions:

1) We propose **SpecMatch-CL**, a spectral graph-matching regularizer that aligns the normalized Laplacian of view-wise embedding graphs and integrates seamlessly with GraphCL-style training.

Preprint.

2) We develop an analysis showing that the spectral loss $\mathcal{L}_G$ simultaneously controls (i) the gap between contrastive loss and its Perfect Alignment counterpart and (ii) the Wang–Isola uniformity objective, which explains how enforcing global spectral alignment benefits both alignment and uniformity in instance-level contrastive learning.

3) We demonstrate consistent improvements in graph classification under unsupervised and semi-supervised regimes and in transfer to molecular / biological property prediction, achieving state-of-the-art results while keeping the training recipe and enhancements unchanged.

2 Related work

2.1 Contrastive learning on graphs

Self-supervised learning on graphs has been largely driven by contrastive objectives that adapt InfoNCE-style losses to node- and graph-level tasks. Early methods such as Deep Graph Infomax (DGI) maximize mutual information between local node representations and a global graph summary, producing strong node-level embeddings without labels [9]. Subsequent work has explored view generation and augmentation at scale. GraphCL adapts InfoNCE loss to the graph domain by applying hand-crafted graph augmentations (node drop, edge perturbation, attribute masking, subgraph sampling) and contrasting two augmented views of each graph, showing that simple GNN backbones with appropriate augmentations already yield strong unsupervised and transfer performance [3]. GRACE and its adaptive variant GCA extend contrastive learning to node-level pretraining through structural and feature corruptions [10, 11]. JOAO also automates the selection of augmentations for GraphCL through a bi-level optimization scheme that chooses augmentations per data set and training step [5]. Across these methods, the primary focus is on instance-level alignment: embeddings of the same node or graph under two augmentations are pulled together, while other examples in the batch serve as negatives. The global geometry of the embedding space—and, in particular, the induced similarity graph over all graphs within a view—remains largely unconstrained.

Recent theoretical work has begun to reinterpret such contrastive objectives in spectral terms. Tan et al. prove that, under standard design choices (normalized embeddings and a Gaussian kernel), minimizing InfoNCE is equivalent to performing spectral clustering on a similarity graph whose edge weights encode positive-pair sampling probabilities [12]. Complementarily, HaoChen et al. show that minimizing the Spectral contrast Loss (SCL), defined on an augmentation graph whose nodes are augmented views and whose edges connect augmentations of the same underlying example, effectively performs a spectral decomposition of this augmentation graph and produces representations with provable linear-probe guaranties [13]. Inspired by these spectral perspectives, our work transfers the idea from an augmentation graph over individual examples to a graph-of-graphs built from graph-level embeddings within each view: we explicitly regularize the normalized Laplacian of the view-specific similarity graphs, rather than relying on InfoNCE alone to shape their global structure.

2.2 Alignment and Uniformity

Wang and Isola analyze InfoNCE through two geometric functionals in the feature distribution: *Alignment* and *Uniformity* in the unit hypersphere [1]. Specifically, denote $z_i, z_j \in \mathbb{R}^d$ as the normalized embeddings (z_i and z_j are in the unit sphere $\mathbb{S}^{d-1}$), and let $p_{\text{pos}}(\cdot)$ and $p_{\text{data}}(\cdot)$ denote the distribution of positive embeddings in $\mathbb{R}^d \times \mathbb{R}^d$ and the embedding distribution in $\mathbb{R}^d$, respectively. The Alignment loss measures the expected distance between embeddings of positive pairs:

$$\mathcal{L}_{\text{align}} \;=\; \mathbb{E}_{(z_i,z_j)\sim p_{\text{pos}}} \big\| z_i - z_j \big\|_2^{\alpha}, \qquad \alpha > 0, \tag{1}$$

and is small when augmentations of the same instance are mapped to nearby points. Uniformity instead quantifies how well the embedding distribution spreads out on the hypersphere via a Gaussian (RBF) potential:

$$\mathcal{L}_{\text{unif}} = \log \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[e^{-t\|z_i - z_j\|_2^2} \right] \tag{2}$$

which is minimized when the embeddings are approximately uniformly distributed in $\mathbb{S}^{d-1}$. Wang and Isola show that, in the limit of many negatives and appropriately chosen temperature, the standard contrastive loss asymptotically optimizes a trade-off between decreasing $\mathcal{L}_{\text{align}}$ (tight clusters for positives) and decreasing $\mathcal{L}_{\text{unif}}$ (globally repulsive, nearly uniform configurations) [1].

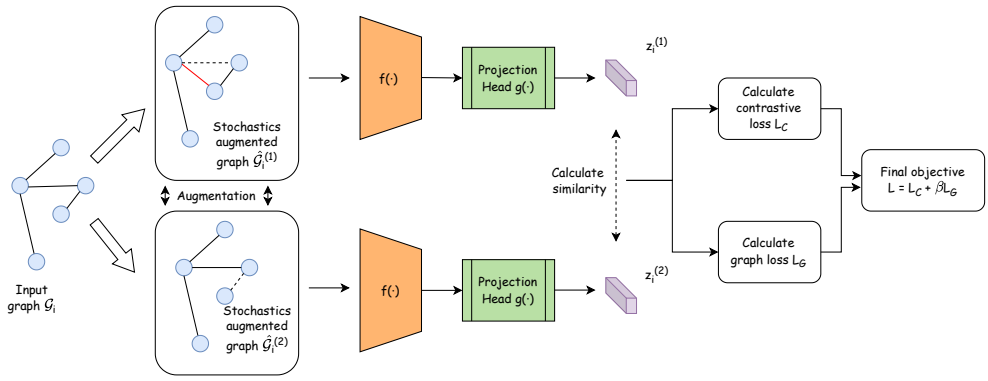

Figure 1: An illustration of SpecMatch-CL method.

Within this framework, they further formalize two idealized regimes. An encoder f is said to achieve *Perfect Alignment* if $z_i = z_j$ almost surely (a.s.) for $(z_i, z_j) \sim p_{\text{pos}}$, i.e., all augmentations of the same instance collapse to a single point on the hypersphere, and it achieves Perfect Uniformity if the distribution of embeddings $p_{\text{data}}(\cdot)$ is the uniform distribution on the unit sphere $\mathbb{S}^{d-1}$. Recent contrastive methods, therefore, minimize both loss to achieve desired performance. Our analysis further shows that reducing the novel spectral graph matching loss $\mathcal{L}_G$ not only tightens an upper bound on the contrastive loss gap to Perfect Alignment but also upper-bounds the Uniformity loss.

3 Method

3.1 Problem definition

In this paper, we focus on graph-level contrastive learning. Let $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i)$ be an undirected graph, where $\mathcal{V}_i = \{v_k\}_{k=1}^{|\mathcal{V}_i|}$ is the set of nodes and $\mathcal{E} = [e_{kj}] \in \mathbb{R}^{|\mathcal{V}_i| \times |\mathcal{V}_i|}$ is the adjacency matrix. Each node v_n is associated with an attribute vector $\mathbf{x}_n \in \mathbb{R}^N$, and we collect them into a feature matrix $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}_i| \times N}$ with $\mathbf{x}_n = \mathbf{X}[n, :]^\top$. In the graph-level setting, we are given a collection of unlabeled graphs

$$\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_N\},$$

and the goal is to learn an encoder that maps each graph $\mathcal{G}_i \in \mathcal{G}$ to a d-dimensional representation $z_i \in \mathbb{R}^d$ using only the structure and features of the graphs.

3.2 Graph augmentations

Following the standard two-view contrastive setup in You et al.[3], for a graph $\mathcal{G}_i \in \mathcal{G}$ we sample two augmented views

$$\hat{\mathcal{G}}_i^{(1)}, \hat{\mathcal{G}}_i^{(2)} \sim \mathcal{T}(\cdot \mid \mathcal{G}_i),$$

where $\mathcal{T}(\cdot \mid \mathcal{G}_i)$ is an augmentation distribution conditioned on $\mathcal{G}_i$, encoding prior assumptions about plausible perturbations of the graph. We consider four basic augmentation operators for constructing positive pairs of graphs:

1. *Node dropping*: randomly remove a subset of nodes together with their incident edges;
2. *Edge perturbation*: randomly add or delete a subset of edges to alter local connectivity;
3. *Attribute masking*: hide a subset of node features and require the encoder to reconstruct or ignore the missing information from context;
4. *Subgraph sampling*: select a subgraph of $\mathcal{G}_i$ using, e.g., a random walk procedure.

3.3 Graph encoder

Once the augmented graph pair $(\hat{\mathcal{G}}_i^{(1)}, \hat{\mathcal{G}}_i^{(2)})$ is generated, we feed each view into a graph encoder to obtain their representations. Our framework is agnostic to the specific architecture, but throughout

we use a graph neural network (GNN) to encode graphs. Specifically, for a view $v \in \{1, 2\}$, let $\hat{\mathcal{G}}_i^{(v)} = (\mathcal{V}_i^{(v)}, \mathcal{E}_i^{(v)})$ with node features $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}_i^{(v)}| \times N}$, where $\mathbf{x}_n = \mathbf{X}[n, :]^\top$ is the feature vector of node $v_n \in \mathcal{V}_i^{(v)}$. Consider an L-layer GNN $f(\cdot)$. The message-passing update at the l-th layer is

$$\mathbf{a}_n^{(l)} = \text{AGGREGATION}^{(l)}\big(\{\mathbf{h}_{n'}^{(l-1)} : n' \in \mathcal{N}(n)\}\big), \tag{3}$$

$$\mathbf{h}_n^{(l)} = \text{COMBINE}^{(l)}\big(\mathbf{h}_n^{(l-1)}, \mathbf{a}_n^{(l)}\big), \tag{4}$$

where $\mathbf{h}_n^{(l)}$ is the embedding of node v_n at layer l, initialized with $\mathbf{h}_n^{(0)} = \mathbf{x}_n$, $\mathcal{N}(n)$ denotes set of nodes adjacent to v_n, and $\text{AGGREGATION}^{(l)}(\cdot)$ and $\text{COMBINE}^{(l)}(\cdot)$ are layer-specific functions (e.g., mean/sum aggregation and nonlinear transformation).

After L layers of propagation, node embeddings are pooled into a graph-level representation via a READOUT function (e.g., sum/mean pooling or attention), then a multi-layer perceptron $g(\cdot)$ is used for downstream graph-level tasks, and the same encoder is shared for all augmented views in the contrastive learning framework. :

$$f(\hat{\mathcal{G}}_i^{(v)}) = \text{READOUT}\big(\{\mathbf{h}_{n'}^{(l-1)} : v_n \in \mathcal{V}_i^{(v)},\, l \in L\}\big),$$
$$z_i^{(v)} = g\left(f(\hat{\mathcal{G}}_i^{(v)})\right).$$

3.4 Contrastive Learning Framework

For the view $v \in \{1, 2\}$, define the other view as v'. As shown by Chen et al. [2] and Khosla et al. [14], normalizing the embeddings helps increase the performance

$$z_i^{(v)} \leftarrow \frac{z_i^{(v)}}{\|z_i^{(v)}\|}, \qquad z_i^{(v')} \leftarrow \frac{z_i^{(v')}}{\|z_i^{(v')}\|}$$

For the normalized embeddings, we define the similarity as inner product

$$s(z_i^{(v)}, z_i^{(v')}) = z_i^{(v)\top} z_i^{(v')}$$

Following the previous works of graph contrastive learning [3] [4] [15], we use normalized temperature-scaled cross-entropy (InfoNCE) loss [16] as the contrastive objective. The single sample contrastive loss according to view v can be written as

$$l_i^{(v)} = -\log \frac{\exp\big(s(z_i^{(v)}, z_i^{(v')})/\tau\big)}{\sum_{a=1}^{2}\sum_{k=1}^{N} \exp\big(s(z_i^{(v)}, z_k^{(a)})/\tau\big) \cdot (1 - \mathbf{1}\{a = v \wedge k = i\})}$$

where $\mathbf{1}(\cdot)$ is the indicator function and τ is the temperature. We then have the total contrastive objective as

$$\mathcal{L}_C \;=\; \sum_{v=1}^{2}\sum_{i=1}^{N} l_i^{(v)}.$$

3.5 Spectral Graph Matching

Given embedding vectors from two views of the data, we optimize the spectral graph matching algorithm by constructs corresponding graphs and compares their spectral properties to ensure structural consistency. Particularly, we construct the similarity matrix $S^{(v)}$ whose entries are defined as:

$$S_{ij}^{(v)} = s(z_i^{(v)}, z_j^{(v)})$$

The adjacency matrices $A^{(v)}$ are then formed by thresholding the similarities:

$$A_{ij}^{(v)} = \begin{cases} 1, & \text{if } S_{ij}^{(v)} > \theta \text{ and } i \neq j \\ 0, & \text{otherwise} \end{cases}$$

with the similarity threshold θ. An example of the adjacency matrices is shown in Figure 2.

Instead of using a fixed threshold, we determine the threshold adaptively based on the distribution of similarity values:

$$\theta = Q(S^{(v)}, p),$$

where $Q(S^{(v)}, p)$ is the p-th percentile of the similarity values in $S^{(v)}$. From each adjacency matrix, we compute the corresponding degree matrices $D^{(v)}$ as

$$D_{ii}^{(v)} = \sum_j A_{ij}^{(v)}$$

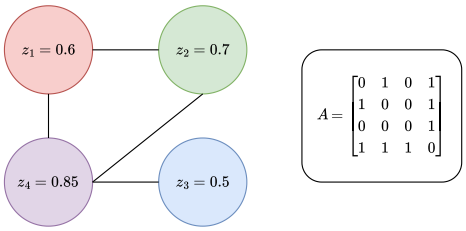

Figure 2: A 1D example of the adjacency matrix A with $\theta = 0.4$.

The normalized Laplacian matrices are defined as:

$$L^{(v)} = I - \left(D^{(v)}\right)^{-\frac{1}{2}} A^{(v)} \left(D^{(v)}\right)^{-\frac{1}{2}},$$

and the spectral graph matching loss is then computed by:

$$\mathcal{L}_G = ||L^{(1)} - L^{(2)}||_F^2.$$

3.6 SpecMatch-CL loss

The total SpecMatch-CL loss combines the contrastive loss and the spectral graph alignment terms:

$$\mathcal{L} = \mathcal{L}_C + \beta \mathcal{L}_G,$$

where α balances the contributions of each loss component. Figure 1 provides an illustration for the process of our method.

4 Theoretical Justification

To theoretically justify SpecMatch-CL, we derive two theorems, built on the Alignment and Uniformity losses introduced by Wang and Isola [1], showing how the spectral graph-matching loss $\mathcal{L}_G$ controls both the gap to the Perfect Alignment case and the Uniformity objective.

4.1 $\mathcal{L}_G$ provides an upper bound for the Contrastive loss gap to Perfect Alignment

In this section, we clarify how the spectral graph-matching loss $\mathcal{L}_G$ influences the contrastive objective. Although InfoNCE operates directly on pairwise similarities between embeddings, $\mathcal{L}_G$ measures discrepancies between view-wise graph geometries encoded by their normalized Laplacians. Our goal is to relate these two levels by showing that if the diffusion geometries of the two views are close, then the realized contrastive loss cannot deviate far from its ideal Perfect Alignment value.

While $\mathcal{L}_G$ penalizes discrepancies between the two view-wise normalized Laplacians, the contrastive objective acts on pairwise similarities between learned embeddings. To connect the two levels of abstraction, we employ the diffusion (heat) kernel [17, 18] as a geometry-aware operator on the graphs induced by each view. Concretely, with a diffusion scale $t_d > 0$, we define the heat kernel for each view v as $P^{(v)} := \exp(-t_d L^{(v)})$, then the associated diffusion distance for each input instance is $\left\|P^{(1)} - P^{(2)}\right\|_F^2$. We posit a mild consistency link between the distances between the two embeddings and the distances their induced diffusion geometries drift. Intuitively, if the embeddings of a positive pair are separate, the neighborhoods they activate in the two views should also look different.

Assumption 4.1. *We assume that there exist a constant c for input graphs $\mathcal{G}$ such that*

$$\sum_{i=1}^{N} \left\|z_i^{(1)} - z_i^{(2)}\right\|_2^2 \leq c \left\|P^{(1)} - P^{(2)}\right\|_F^2$$

This implies that diffusion mismatch upper-bounds embedding mismatch up to a data-dependent factor c. The assumption is violated in the degenerate case $\left\|P^{(1)} - P^{(2)}\right\|_F^2 = 0$, i.e., when $P^{(1)} = P^{(2)}$ (equivalently $L^{(1)} = L^{(2)}$) and the two view-induced graphs are identical, which is rare in practice. With that assumption, we introduce the theorem:

Theorem 4.2. *Under Assumption 4.1 we have*

$$|\mathcal{L}_C - \mathcal{L}_C^*| \leq \frac{(t_d)^2 c}{\tau} \mathcal{L}_G$$

a.s. over $(z_i^{(1)}, z_i^{(2)}) \sim p_{pos}$, where τ is the temperature in contrastive loss and t_d is the diffusion scale.

The proof for Theorem 4.2 is provided in Appendix A.

Discussion. Under Assumption 4.1, the theorem establishes a link between the spectral alignment and the contrastive objective: a reduction in the spectral graph-matching loss $\mathcal{L}_G$ produces a provably tighter upper bound on the deviation between the realized contrastive loss and its Perfect Alignment counterpart. This observation provides a rationale for adding the graph matching loss: by constraining the view-wise Laplacians to be close, not only the geometry of the two views are aligned but also the contrastive objective becomes closer to its ideal value, which is consistent with improvements in unsupervised, semi-supervised and transfer accuracy as shown in the experiment section.

4.2 $\mathcal{L}_G$ provides an upper bound for the Uniformity loss

Having established that the spectral graph-matching loss $\mathcal{L}_G$ controls the deviation from the Perfect Alignment contrastive objective, we now turn to its effect on uniformity. Recall that, in the Wang–Isola framework, uniformity quantifies how well the embedding distribution spreads out on the unit sphere, penalizing collapsed or highly clustered configurations. Our goal here is to show that enforcing a small spectral discrepancy between view-wise Laplacians also drives the encoder toward low uniformity loss, i.e., toward a more evenly dispersed configuration of graph-level embeddings. Intuitively, if the two views induce similar diffusion geometries on the same node set, then their embeddings cannot concentrate in a few narrow regions without incurring a large Laplacian mismatch.

Assume that each augmentation graph is connected; denote $d_i^{(v)} := D_{ii}^{(v)}$ as the i^{th} diagonal elements of $D^{(v)}$ and λ_2 as the smallest non-zero eigenvalue of the normalized Laplacian matrix L. In addition, let the distribution of L (over the randomness of the augmentations) be conditional on the set of input graphs $\mathcal{G}$ as $\mathbb{P}_{L|\mathcal{G}}$. Additionally, denote $\overline{L} := \mathbb{E}_{L \sim \mathbb{P}_{L|\mathcal{G}}}[L]$ and $\overline{\lambda}_2$ as the smallest non-zero eigenvalue of $\overline{L}$. Recall the Wang–Isola uniformity potential at temperature $t > 0$:

$$\mathcal{L}_{\text{unif}} = \log \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[e^{-t\|z_i - z_j\|_2^2} \right].$$

Let Z be the matrix of embeddings used to construct L. For embeddings z_i (rows of Z), let the degree-weighted mean be

$$\mu := \frac{1}{\sum_{k=1}^N d_k} \sum_{i=1}^N d_i z_i.$$

Theorem 4.3. *Assume that each augmentation graph is connected and $L^{(1)}$ and $L^{(2)}$ are i.i.d. (conditional on $\mathcal{G}$). We have*

$$\mathcal{L}_{\text{unif}} \leq \frac{1 - e^{-4t}}{2\sqrt{2}} \left(\frac{3}{2} - \mathbb{E}\left[\|\mu\|_2^2\right] \right) \sqrt{\mathbb{E}_{L^{(1)}, L^{(2)} \sim \mathbb{P}_{L|\mathcal{G}}}[\mathcal{L}_G]} - \frac{(1 - e^{-4t})}{2} \overline{\lambda}_2 (1 - \mathbb{E}\left[\|\mu\|_2^2\right]),$$

where t is the temperature parameter of the Uniformity loss.

The proof for Theorem 4.3 is provided in Appendix B.

Discussion. Theorem 4.3 links our spectral regularizer directly to the Wang–Isola uniformity objective. The first term on the right-hand side shows that $\mathcal{L}_{\text{unif}}$ grows at most like $\sqrt{\mathbb{E}[\mathcal{L}_G]}$, so reducing the spectral graph-matching loss tightens a nontrivial upper bound on uniformity: better spectral alignment between views forces the embedding distribution to be more spread out. The term $|\mu|_2^2$ is the squared norm of the degree-weighted mean embedding: when representations are well balanced on the sphere, $|\mu|_2^2$ is small and $(1 - \mathbb{E}\left[|\mu|_2^2\right])$ is close to one, strengthening the negative contribution of the second term; when the encoder collapses the mass into a few directions, $|\mu|_2^2$ grows and the bound weakens. The dependence on $\overline{\lambda}_2$, the smallest non-zero eigenvalue of the expected normalized Laplacian, ties uniformity to graph connectivity: a larger spectral gap (better-connected similarity graph) tightens the bound and promotes more uniform embeddings.

We empirically assess the impact of the spectral graph-matching loss by tracking checkpoints of contrastive training with and without this term every two epochs when training on NCI1 dataset and plotting their alignment and uniformity losses in Figure 3. In this experiment, we use $\alpha = 2$ for $\mathcal{L}_{\text{align}}$, $t = 2$ for $\mathcal{L}_{\text{unif}}$, and $\beta = 0.5$. While both variants progressively improve $\mathcal{L}_{\text{align}}$ and $\mathcal{L}_{\text{unif}}$, the model with the spectral graph-matching loss achieves consistently faster reductions in both metrics. Taken together, the theorems show that minimizing $\mathcal{L}_G$ simultaneously pulls the contrastive loss toward its Perfect Alignment limit and acts as a spectral surrogate for the uniformity criterion, providing a unified explanation for the empirical gains observed across unsupervised, semi-supervised, and transfer settings.

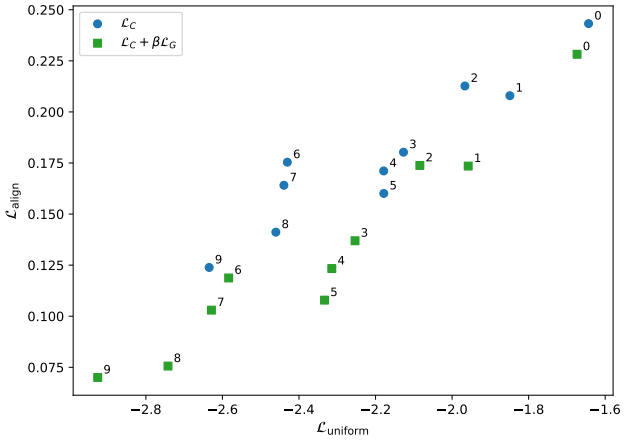

Figure 3: $\mathcal{L}_{align} - \mathcal{L}_{unif}$ plot for contrastive learning with and without the spectral graph matching loss on NCI1 dataset. The numbers around the points are the indexes of epochs. For both $\mathcal{L}_{align}$ and $\mathcal{L}_{unif}$, lower is better.

5 Experiments

We evaluate SpecMatch-CL to test whether spectral graph matching consistently improves over strong contrastive baselines, covering three regimes: unsupervised learning, semi-supervised learning (both are conducted on TU benchmarks), and transfer learning from large-scale pre-training to molecular and biological prediction tasks.

5.1 Experimental Setup

Training framework and hyperparameters We adopt the GraphCL training framework as the base [3]. We follow its default augmentation strength (0.2) and choose augmentation operators by data regime: for biochemical molecules, we apply node dropping and subgraph extraction; for dense social networks, we use all four operators; and for sparse social networks, we use all except attribute masking. For our adaptive similarity threshold, we set the percentile to $p = 80$ by default (ablation reported in Appendix C). We align our backbones and hyperparameters with widely used settings to ensure comparability across regimes. (1) *Unsupervised representation learning*: we use GIN with 3 layers and 32 hidden dimensions [19]. (2) *Semi-supervised learning*: we use a 5-layer ResGCN with 128 hidden dimensions [20]. (3) *Transfer learning*: we use GIN with 5 layers and 300 hidden dimensions, following standard practice [21]. Unless specified, the weight of our spectral loss is $\beta \in \{0.5, 0.75, 1.0, 1.25, 1.5\}$, selected by grid search (for ablation study of β, see Appendix D).

Datasets For unsupervised and semi-supervised graph-level learning, we follow standard practice and evaluate on the TU benchmark collection [22], which comprises social network graphs: COLLAB,

Table 1: Graph classification accuracy (%) on benchmark datasets under unsupervised training. Best results are presented in bold

Method	NCI1	PROTEINS	DD	MUTAG	COLLAB	RDT-B	RDT-M5K	IMDB-B
WL	80.01±0.50	72.92±0.56	74.02±2.28	80.72±3.00	60.30±3.44	68.82±0.41	46.06±0.21	72.30±3.44
DGK	80.31±0.46	73.30±0.82	74.85±0.74	87.44±2.72	64.66±0.50	78.04±0.39	41.27±0.18	66.96±0.56
sub2vec	52.84±1.47	53.03±5.55	54.33±2.44	61.05±15.80	55.26±1.54	71.48±0.41	36.68±0.42	55.26±1.54
node2vec	54.89±1.61	57.49±3.57	74.77±0.51	72.63±10.20	54.57±0.37	72.76±0.92	31.09±0.14	58.02±2.30
graph2vec	73.22±1.81	73.30±2.05	70.32±2.32	83.15±9.25	71.10±0.54	75.48±1.03	47.86±0.26	71.10±0.54
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
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
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±0.38
JOAO V2	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
SimGRACE	79.12±0.44	75.35±0.09	77.44±1.11	89.01±1.31	71.72±0.82	89.51±0.89	55.91±0.34	71.26±0.74
MSSGCL	81.45±0.48	75.49±0.70	79.73±0.44	89.68±0.57	73.48±0.83	91.08±0.78	56.17±0.18	73.14±0.38
CuCo	79.24±0.56	75.91±0.55	79.20±1.12	90.55±0.98	72.30±0.34	88.6±0.55	56.49±0.19	72.33±0.22
SpecMatch-CL	**81.86**±0.36	**76.69**±0.50	**81.12**±1.04	**90.87**±0.75	**74.26**±0.94	**91.31**±0.79	**57.22**±1.20	**73.35**±0.44

IMDB-B/M, RDT-B, RDT-M5K, commonly traced to graph kernel benchmarks [23, 24], as well as biochemical molecule datasets: MUTAG, NCI1, PROTEINS, DD. For transfer learning, we pre-train on large unlabeled dataset ZINC 2M (from the ZINC database) [25] and PPI-306K (as in graph pre-training protocols) [26]—and fine-tune on downstream suites: molecular property prediction tasks from MoleculeNet: BBBP, ToxCast, SIDER, ClinTox, MUV, HIV, BACE [27], and protein–protein interaction classification dataset PPI [28].

Evaluation protocols Following widely adopted evaluation protocols for graph-level self-supervised learning [7, 3, 5, 6], we assess generalization in both unsupervised and semi-supervised regimes. In the unsupervised setting, we train SpecMatch-CL on the full training graphs to obtain graph-level embeddings and then fit a downstream linear SVM with 10-fold cross-validation on each dataset [29]. In the semi-supervised setting, we pre-train GNNs with SpecMatch-CL on all available training graphs and fine-tune with the prescribed label-rate protocol (stratified K-fold when explicit splits are unavailable; otherwise train/val/test splits as provided by the benchmark). Hyperparameters for fine-tuning are selected on validation sets; we report mean and standard deviation over multiple random seeds.

Baselines We compare SpecMatch-CL against (i) classical graph-kernel methods: Weisfeiler–Lehman (WL) [24] and Deep Graph Kernels (DGK) [23]; (ii) unsupervised graph embedding baselines—sub2vec [30], node2vec [31], and graph2vec [29]; and (iii) representative graph contrastive/self-supervised methods—InfoGraph [7], GraphCL [3], JOAO and JOAOv2 [5, 6], SimGRACE [32], Multi-Scale Subgraph Contrastive Learning (MSSGCL) [15], and CuCo (Curriculum Contrastive Learning) [4]. All baselines are trained under their recommended settings to ensure comparability.

5.2 Unsupervised training

As shown in Table 1, under the same GraphCL training recipe and augmentation strength, SpecMatch-CL attains state-of-the-art accuracy on all eight TU benchmarks. Compared with the strongest prior method on each dataset, it improves performance by about 0.61 percentage points on average (e.g., $+0.41$ on NCI1, $+0.78$ on PROTEINS, $+1.39$ on DD, $+0.32$ on MUTAG, $+0.78$ on COLLAB, $+0.23$ on RDT-B, $+0.73$ on RDT-M5K, and $+0.21$ on IMDB-B). These consistent gains indicate that enforcing view-to-view spectral alignment provides complementary benefits to standard instance-level objectives and augmentation design, leading to uniformly stronger graph-level representations under unsupervised pretraining.

5.3 Semi-supervised training

Table 2 shows that at the 1% label rate, SpecMatch-CL surpasses the strongest baselines on both datasets with available splits, reaching 65.12 on NCI1 and 65.86 on COLLAB, which corresponds to improvements of $+0.49$ and $+0.84$ points over MSSGCL, respectively. At the 10% label rate, SpecMatch-CL achieves the best results on 5 out of 6 datasets: NCI1 (75.67, $+0.81$ vs. Infomax/JOAOv2 at 74.86), DD (79.21, $+0.32$ vs. MSSGCL), COLLAB (76.55, $+0.53$ vs. MSSGCL),

Table 2: Results (%) on semi-supervised graph classification. "−" indicates that label rate is too low for the given dataset size

LR	Methods	NCI1	PROTEINS	DD	COLLAB	RDT-B	RDT-M5K
1%	No pre-train.	60.72 ± 0.45	–	–	57.46 ± 0.25	–	–
	Augmentations	60.49 ± 0.46	–	–	58.40 ± 0.97	–	–
	GAE	61.63 ± 0.84	–	–	63.20 ± 0.67	–	–
	Infomax	62.72 ± 0.65	–	–	61.70 ± 0.77	–	–
	ContextPred	61.21 ± 0.77	–	–	57.60 ± 2.07	–	–
	GraphCL	62.55 ± 0.86	–	–	64.57 ± 1.15	–	–
	JOAO	61.97 ± 0.72	–	–	63.71 ± 0.84	–	–
	JOAOv2	62.52 ± 1.16	–	–	64.51 ± 2.21	–	–
	SimGRACE	64.21 ± 0.65	–	–	64.28 ± 0.98	–	–
	MSSGCL	64.63 ± 0.75	–	–	65.02 ± 0.78	–	–
	SpecMatch-CL	**65.12** ± 0.65	–	–	**65.86**± 0.98	–	–
10%	No pre-train.	73.72 ± 0.24	70.40 ± 1.54	73.56 ± 0.41	73.71 ± 0.27	86.63 ± 0.27	51.33 ± 0.44
	Augmentations	73.59 ± 0.32	70.29 ± 0.64	74.30 ± 0.81	74.19 ± 0.13	87.74 ± 0.39	52.01 ± 0.20
	GAE	74.36 ± 0.24	70.51 ± 0.17	74.54 ± 0.68	75.09 ± 0.19	87.69 ± 0.40	33.58 ± 0.13
	Infomax	74.86 ± 0.26	72.27 ± 0.40	75.78 ± 0.34	73.76 ± 0.29	88.66 ± 0.95	53.61 ± 0.31
	ContextPred	73.00 ± 0.30	70.23 ± 0.63	74.66 ± 0.51	73.69 ± 0.37	84.76 ± 0.52	51.23 ± 0.84
	GraphCL	74.63 ± 0.25	74.17 ± 0.34	76.17 ± 1.37	74.23 ± 0.21	89.11 ± 0.19	52.55 ± 0.45
	JOAO	74.48 ± 0.27	72.13 ± 0.92	75.69 ± 0.67	75.30 ± 0.32	88.14 ± 0.25	52.83 ± 0.54
	JOAOv2	74.86± 0.39	73.31 ± 0.48	75.81 ± 0.73	75.53 ± 0.18	88.79 ± 0.65	52.71 ± 0.28
	SimGRACE	74.60 ± 0.41	74.03 ± 0.51	76.48 ± 0.52	74.74 ± 0.28	88.86 ± 0.62	53.97 ± 0.64
	MSSGCL	74.77 ± 0.31	**75.76** ± 0.52	78.89 ± 0.18	76.02 ± 0.13	90.58 ± 0.34	54.36 ± 0.24
	SpecMatch-CL	**75.67** ± 0.31	75.06 ± 0.68	**79.21** ± 1.12	**76.55** ± 0.34	**91.86** ± 0.42	**55.26**± 0.35

Table 3: Transfer learning results (ROC-AUC %) on benchmark datasets.

Pre-Train dataset	PPI-306K	ZINC 2M							
Pre-Train dataset	PPI	Tox21	ToxCast	Sider	ClinTox	MUV	HIV	BBBP	Bace
No Pre-Train	64.8(1.0)	74.6(0.4)	61.7(0.5)	58.2(1.7)	58.4(6.4)	70.7(1.8)	75.5(0.8)	65.7(3.3)	72.4(3.8)
EdgePred	65.7(1.3)	76.0(0.6)	64.1(0.6)	60.4(0.7)	64.1(3.7)	75.1(1.2)	76.3(1.0)	67.3(2.4)	77.3(3.5)
AttrMasking	65.2(1.6)	75.1(0.9)	63.3(0.9)	60.5(0.9)	73.5(4.3)	75.8(1.0)	75.3(1.5)	65.2(1.4)	77.8(1.8)
ContextPred	64.4(1.3)	73.6(0.3)	62.6(0.6)	59.7(1.8)	74.0(3.4)	72.5(1.5)	75.6(1.0)	70.6(1.5)	**78.8**(1.2)
GraphCL	67.88(0.85)	75.1(0.7)	63.0(0.4)	59.8(1.3)	77.5(3.8)	76.4(0.4)	75.1(0.7)	67.8(2.4)	74.6(2.1)
JOAO	64.43(1.38)	74.8(0.6)	62.8(0.7)	60.4(1.5)	66.6(3.1)	76.6(1.7)	**76.9**(0.7)	66.4(1.0)	73.2(1.6)
SimGRACE	70.25(1.22)	75.6(0.5)	63.4(0.5)	60.6(1.0)	75.6(3.0)	76.9(1.3)	75.2(0.9)	71.3(0.9)	75.0(1.7)
SpecMatch-CL	**71.75**(0.82)	**76.97**(0.45)	**64.22**(0.45)	**62.44**(1.22)	**77.78**(3.2)	**78.86**(1.37)	76.25(0.8)	**72.88**(1.2)	76.93(1.8)

RDT-B (91.86, +1.28 vs. MSSGCL), and RDT-M5K (55.26, +0.90 vs. MSSGCL), while remaining competitive on PROTEINS (75.06 vs. 75.76 for MSSGCL). These results suggest that enforcing view-to-view spectral consistency is particularly beneficial when labels are limited and when preserving multi-hop neighborhood structure is critical, while maintaining strong performance in settings where existing augmentations already regularize the geometry.

5.4 Transfer learning

Pre-training with SpecMatch-CL transfers strongly across biochemistry and biology tasks (Table 3), attaining the best ROC-AUC on 7 out of 9 downstream datasets and delivering an average gain of roughly +0.64 points over the strongest baseline per dataset. Improvements are substantial on PPI (71.75 vs. 70.25, +1.50), Tox21 (76.97 vs. 76.00, +0.97), SIDER (62.44 vs. 60.60, +1.84), MUV (78.86 vs. 76.90, +1.96), and BBBP (72.88 vs. 71.30, +1.58), with a more modest gain on ToxCast (64.22 vs. 64.10, +0.12) and ClinTox (77.78 vs. 77.50, +0.28). Performance is competitive on HIV (within 0.65 of the best score of 76.9) and trails on BACE (76.93 vs. 78.8, −1.87), suggesting that endpoint-specific structure (e.g., motif sensitivity) may warrant tuning of the spectral loss and diffusion parameters on certain targets.

6 Conclusion

We presented **SpecMatch-CL**, a simple and effective loss that enforces view-to-view spectral alignment in graph contrastive learning. By matching the spectra of normalized Laplacians, the method preserves multi-scale neighborhood structure across augmentations and complements the alignment–uniformity trade-off optimized by InfoNCE. Our diffusion-kernel analysis further shows that

the spectral graph-matching loss $\mathcal{L}_G$ simultaneously controls the gap to the Perfect Alignment contrastive objective and upper-bounds the Wang–Isola Uniformity loss, yielding a model-agnostic theoretical justification for the graph-matching loss. Empirically, SpecMatch-CL delivers consistent improvements on unsupervised and semi-supervised TU benchmarks and strengthens transfer performance on diverse molecular and biological datasets, all within a standard GraphCL training pipeline. Limitations include sensitivity to graph-construction choices (e.g., similarity threshold) and to the spectral graph-matching loss weight β, although we observe broad robustness across datasets in practice. Future directions include adaptive scheduling of the graph-matching loss weight, extensions to node-level and heterogeneous graphs, alternative spectral penalties (e.g., Ky–Fan norms), and multi-view generalizations that jointly align more than two augmented graphs.

7 Acknowledgement

The author thanks Joshua Cape for helpful comments for helpful comments on an early draft on this paper.

References

[1] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9929–9939. PMLR, 2020.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020.

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

[4] Guanyi Chu, Xiao Wang, Chuan Shi, and Xunqiang Jiang. CuCo: Graph representation with curriculum contrastive learning. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, IJCAI-21, pages 2300–2306. IJCAI, 2021.

[5] Yuning You, Tianlong Chen, Yang Shen, and Zhangyang Wang. JOAO: Graph contrastive learning with automated augmentations. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12121–12132. PMLR, 2021.

[6] Yuning You, Tianlong Chen, Yongduo Wang, Yang Shen, Zhangyang Wang, and Yang Yang. JOAOv2: Minimalist automated augmentations for graph contrastive learning. *arXiv preprint arXiv:2107.02024*, 2021.

[7] Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. InfoGraph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *International Conference on Learning Representations*, 2020.

[8] Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *Proceedings of the 37th International Conference on Machine Learning*, ICML, 2020.

[9] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R. Devon Hjelm. Deep graph infomax. In *International Conference on Learning Representations*, 2019.

[10] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep graph contrastive representation learning. In *ICML Workshop on Graph Representation Learning and Beyond*, 2020.

[11] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference*, WWW, pages 2069–2080, 2021.

[12] Zhiquan Tan, Yifan Zhang, Jingqin Yang, and Yang Yuan. Contrastive learning is spectral clustering on similarity graph. In *International Conference on Learning Representations*, ICLR, 2024.

[13] Jeff Z. HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. In *Advances in Neural Information Processing Systems*, volume 34, pages 5000–5012, 2021.

[14] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673, 2020.

[15] Yanbei Liu, Yu Zhao, Xiao Wang, Lei Geng, and Zhitao Xiao. Multi-scale subgraph contrastive learning. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence*, IJCAI-23, pages 2215–2223. IJCAI, 2023.

[16] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. In *Workshop on Self-Supervised Learning, Advances in Neural Information Processing Systems (NeurIPS)*, 2018. arXiv:1807.03748.

[17] Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006.

[18] Peter W. Jones, Mauro Maggioni, and Raanan Schul. Manifold parametrizations by eigenfunctions of the laplacian and heat kernels. *Proceedings of the National Academy of Sciences*, 105(6):1803–1808, 2008.

[19] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

[20] Ting Chen, Song Bian, and Yizhou Sun. Are powerful graph neural nets necessary? a dissection on graph classification. *CoRR*, abs/1905.04579, 2019.

[21] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems*, volume 33, pages 22118–22133, 2020.

[22] Christopher Morris, Nils M. Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. TUDataset: A collection of benchmark datasets for learning with graphs. In *ICML 2020 Workshop on Graph Representation Learning and Beyond (GRL+ 2020)*, 2020.

[23] Pinar Yanardag and S. V. N. Vishwanathan. Deep graph kernels. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1365–1374. ACM, 2015.

[24] Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn, and Karsten M. Borgwardt. Weisfeiler–lehman graph kernels. *Journal of Machine Learning Research*, 12:2539–2561, 2011.

[25] Teague Sterling and John J. Irwin. ZINC 15: Ligand discovery for everyone. *Journal of Chemical Information and Modeling*, 55(11):2324–2337, 2015.

[26] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations*, 2020.

[27] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. Moleculenet: A benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018.

[28] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, pages 1025–1035, 2017.

[29] Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, Yang Liu, and Shantanu Jaiswal. graph2vec: Learning distributed representations of graphs. In *Proceedings of the 13th International Workshop on Mining and Learning with Graphs (MLG)*, 2017.

[30] Bijaya Adhikari, Yao Zhang, Naren Ramakrishnan, and B. Aditya Prakash. Sub2vec: Feature learning for subgraphs. In *Advances in Knowledge Discovery and Data Mining*, PAKDD, pages 170–182. Springer, 2018.

[31] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM, 2016.

[32] Jun Xia, Lirong Wu, Jintao Chen, Bozhen Hu, and Stan Z. Li. SimGRACE: A simple framework for graph contrastive learning without data augmentation. In *Proceedings of the ACM Web Conference 2022*, WWW '22, pages 1070–1079. ACM, 2022.

[33] Amnon Pazy. *Semigroups of Linear Operators and Applications to Partial Differential Equations*, volume 44 of *Applied Mathematical Sciences*. Springer, New York, 1983.

[34] A. J. Hoffman and H. W. Wielandt. The variation of the spectrum of a normal matrix. *Duke Mathematical Journal*, 20(1):37–39, 1953.

[35] Rajendra Bhatia. *Matrix Analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer, New York, 1997.

A Proof for Theorem 4.2

Since the Laplacian matrices are symmetric positive semi-definite, we have $\|e^{-uL^{(1)}}\|_2 \leq 1$ and $\|e^{-uL^{(2)}}\|_2 \leq 1$ for all $u > 0$. We now prove that

$$\|P^{(1)} - P^{(2)}\|_F^2 \leq (t_d)^2 \cdot \mathcal{L}_G$$

using the Duhamel formula [33].
Let $G(k) = e^{-(t_d-k)L^{(1)}} e^{-kL^{(2)}}$. We then have $G(0) = e^{-t_d L^{(1)}}$ and $G(t_d) = e^{-t_d L^{(2)}}$. Differentiate:

$$\begin{aligned}
&\frac{d}{dk}G(k) = e^{-(t_d-k)L^{(1)}}(L^{(1)} - L^{(2)})e^{-kL^{(2)}} \\
\Rightarrow &e^{-t_d L^{(2)}} - e^{-t_d L^{(1)}} = \int_0^{t_d} e^{-(t_d-k)L^{(1)}}(L^{(1)} - L^{(2)})e^{-kL^{(2)}}\, dk
\end{aligned}$$

By submultiplicativity of matrix norms we get

$$\left\|P^{(1)} - P^{(2)}\right\|_F = \|e^{-t_d L^{(1)}} - e^{-t_d L^{(2)}}\|_F \leq \int_0^{t_d} \|e^{-(t_d-k)L^{(1)}}\|_2 \, \|L^{(1)} - L^{(2)}\|_F \, \|e^{-kL^{(2)}}\|_2 \, \mathrm{d}k$$

Since $\|e^{-(t_d-k)L^{(1)}}\|_2 \leq 1$ and $\|e^{-kL^{(2)}}\|_2 \leq 1$, we have

$$\begin{aligned}
\left\|P^{(1)} - P^{(2)}\right\|_F &\leq \|L^{(1)} - L^{(2)}\|_F \int_0^{t_d} 1 \, \mathrm{d}k \\
&= t_d \, \|L^{(1)} - L^{(2)}\|_F,
\end{aligned}$$

which means

$$\left\|P^{(1)} - P^{(2)}\right\|_F^2 \leq (t_d)^2 \cdot \mathcal{L}_G$$

Because the embedding vectors are normalized, we have

$$s(z_i^{(1)}, z_i^{(2)}) = 1 - \frac{1}{2}\|z_i^{(1)} - z_i^{(2)}\|_2^2$$

Let $z_i^{*(1)}$ and $z_i^{*(2)}$ be the embedding vectors in the Perfect Alignment case. When $z_i^{*(1)} = z_i^{*(2)}$, we have $\|z_i^{*(1)} - z_i^{*(2)}\|_2^2 = 0$.

Since all embedding vectors are unit-norm, if $z_i^{*(1)} = z_i^{*(2)}$ we have

$$\begin{aligned}
\left|s(z_i^{(1)}, z_i^{(2)}) - s(z_i^{*(1)}, z_i^{*(2)})\right| &= \left|1 - \frac{1}{2}\|z_i^{(1)} - z_i^{(2)}\|_2^2 - 1 + \frac{1}{2}\|z_i^{*(1)} - z_i^{*(2)}\|_2^2\right| \\
&= \left|0 - \frac{1}{2}\|z_i^{(1)} - z_i^{(2)}\|_2^2 + 0\right| \\
&= \frac{1}{2}\|z_i^{(1)} - z_i^{(2)}\|_2^2
\end{aligned}$$

Note that we can write the single sample contrastive loss as

$$
\begin{aligned}
l_i^{(v)} &= -\log \frac{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)}{\sum_{a=1}^{2}\sum_{k=1}^{N}\exp\big(s(z_i^{(v)}, z_k^{(a)})/\tau\big)\cdot(1-\mathbf{1}\{a=v\wedge k=i\})} \\
&= -\log \frac{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)}{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)+\sum_{a=1}^{2}\sum_{k=1}^{N}\exp\big(s(z_i^{(v)}, z_k^{(a)})/\tau\big)\cdot(1-\mathbf{1}\{k=i\})} \\
&= -\frac{s\big(z_i^{(v)}, z_i^{(v')}\big)}{\tau} + \log\left(\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)+\sum_{a=1}^{2}\sum_{k=1}^{N}\exp\big(s(z_i^{(v)}, z_k^{(a)})/\tau\big)\cdot(1-\mathbf{1}\{k=i\})\right)
\end{aligned}
$$

Taking the derivative of $l_i^{(v)}$ with respect to $s(z_i^{(v)}, z_i^{(v')})$ gives:

$$
\frac{\partial l_i^{(v)}}{\partial s(z_i^{(v)}, z_i^{(v')})} = \frac{1}{\tau}\left(\frac{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)}{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)+\sum_{a=1}^{2}\sum_{k=1}^{N}\exp\big(s(z_i^{(v)}, z_k^{(a)})/\tau\big)\cdot(1-\mathbf{1}\{k=i\})}-1\right)
$$

Note that

$$
\frac{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)}{\exp(s(z_i^{(v)}, z_i^{(v')})/\tau)+\sum_{a=1}^{2}\sum_{k=1}^{N}\exp\big(s(z_i^{(v)}, z_k^{(a)})/\tau\big)\cdot(1-\mathbf{1}\{k=i\})} \in [0,1],
$$

which means

$$
\left|\frac{\partial l_i^{(v)}}{\partial s(z_i^{(v)}, z_i^{(v')})}\right| \le \frac{1}{\tau}
$$

So $l_i^{(v)}$ is $\frac{1}{\tau}$-Lipschitz in $s(z_i^{(v)}, z_i^{(v')})$. If $z_i^{*(1)} = z_i^{*(2)}$ we have

$$
\begin{aligned}
\left|l_i^{(v)} - l_i^{*(v)}\right| &\le \frac{1}{\tau}\left|s(z_i^{(v)}, z_i^{(v')}) - s(z_i^{*(v)}, z_i^{*(v')})\right| \\
&= \frac{1}{2\tau}\|z_i^{(1)} - z_i^{(2)}\|_2^2
\end{aligned}
$$

Therefore

$$
\begin{aligned}
|\mathcal{L}_C - \mathcal{L}_C^*| &= \left|\sum_{v=1}^{2}\sum_{i=1}^{N} l_i^{(v)} - \sum_{v=1}^{2}\sum_{i=1}^{N} l_i^{*(v)}\right| \\
&\le \sum_{v=1}^{2}\sum_{i=1}^{N}\left|l_i^{(v)} - l_i^{*(v)}\right| \\
&\text{(by Triangle inequality)} \\
&\le \sum_{i=1}^{N}\frac{2}{2\tau}\|z_i^{(1)} - z_i^{(2)}\|_2^2 \\
&\le \frac{(t_d)^2\ 2c}{2\tau}\mathcal{L}_G \\
&= \frac{(t_d)^2 c}{\tau}\mathcal{L}_G
\end{aligned}
$$

a.s. over $(z_i^{(1)}, z_i^{(2)}) \sim p_{pos}$.

B Proof for Theorem 4.3

By the independent assumption

$$\begin{aligned}
\mathbb{E}_{L^{(1)},L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}[\mathcal{L}_G] &= \mathbb{E}_{L^{(1)},L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L^{(1)}-L^{(2)}\|_F^2\right] \\
&= \mathbb{E}_{L^{(1)},L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L^{(1)}-\overline{L}-L^{(2)}+\overline{L}\|_F^2\right] \\
&= \mathbb{E}_{L^{(1)},L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L^{(1)}-\overline{L}\|_F^2+\|L^{(2)}-\overline{L}\|_F^2-2\operatorname{tr}\left((L^{(1)}-\overline{L})^\top(L^{(2)}-\overline{L})\right)\right] \\
&= \mathbb{E}_{L^{(1)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L^{(1)}-\overline{L}\|_F^2\right]+\mathbb{E}_{L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L^{(2)}-\overline{L}\|_F^2\right]- \\
&\quad 2\sum_{i,j}\mathbb{E}_{L^{(1)},L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[(L^{(1)}-\overline{L})_{ij}(L^{(2)}-\overline{L})_{ij}\right] \\
&= 2\mathbb{E}_{L\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L-\overline{L}\|_F^2\right]- \\
&\quad 2\sum_{i,j}\mathbb{E}_{L^{(1)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[(L^{(1)}-\overline{L})_{ij}\right]\mathbb{E}_{L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}\left[(L^{(2)}-\overline{L})_{ij}\right] \\
&= 2\mathbb{E}_{L\sim\mathbb{P}_{L|\mathcal{G}}}\left[\|L-\overline{L}\|_F^2\right].
\end{aligned}$$

By Hoffman–Wielandt inequality [34], we have

$$\left(\lambda_2-\overline{\lambda}_2\right)^2\le\|L-\overline{L}\|_F^2,$$

which means

$$\mathbb{E}_{L\sim\mathbb{P}_{L|\mathcal{G}}}\left[(\lambda_2-\overline{\lambda}_2)^2\right]\le\frac{1}{2}\mathbb{E}_{L^{(1)},L^{(2)}\sim\mathbb{P}_{L|\mathcal{G}}}[\mathcal{L}_G].$$

For any x,

$$\begin{aligned}
x^\top Lx &= x^\top x-x^\top D^{-1/2}AD^{-1/2}x \\
&= \frac{1}{2}\sum_{i,j}A_{ij}\left(\frac{x_i}{\sqrt{d_i}}-\frac{x_j}{\sqrt{d_j}}\right)^2.
\end{aligned}$$

Let $u_1=\dfrac{D^{1/2}\mathbf{1}}{\|D^{1/2}\mathbf{1}\|}$ (the eigenvector of L for eigenvalue 0). By Rayleigh-Ritz theorem (see [35]), we have

$$\begin{aligned}
\lambda_2 &= \min_{\substack{x\neq0\\x\perp u_1}}\frac{x^\top Lx}{\|x\|_2^2} \\
\Rightarrow x^\top Lx &\ge \lambda_2\|x\|_2^2\quad\forall x\perp u_1.
\end{aligned}$$

Let Z be the matrix of embeddings used to construct L. For all columns $Z_{:,r}$ of Z we then define

$$\begin{aligned}
G_{:,r} &:= Z_{:,r}-\mu_r\mathbf{1}, \\
x_r &:= D^{\frac{1}{2}}G_{:,r}.
\end{aligned}$$

Notice that

$$\begin{aligned}
x_r^\top u_1 &= \frac{1}{\|D^{1/2}\mathbf{1}\|}\left(D^{\frac{1}{2}}(Z_{:,r}-\mu_r\mathbf{1})\right)^\top D^{1/2}\mathbf{1} \\
&= \frac{1}{\|D^{1/2}\mathbf{1}\|}\sum_{i=1}^N d_i(z_{i,r}-\mu_r) \\
&= \frac{1}{\|D^{1/2}\mathbf{1}\|}\left(\sum_{i=1}^N d_iz_{i,r}-\mu_r\sum_{i=1}^N d_i\right)=0,
\end{aligned}$$

which means

$$
\begin{aligned}
& x_r^\top L x_r \geq \lambda_2 \|x_r\|_2^2, \\
\Rightarrow & \frac{1}{2} \sum_{i,j} A_{ij} \left(G_{i,r} - G_{j,r}\right)^2 \geq \lambda_2 \sum_i d_i (G_{i,r})^2 \\
\Rightarrow & \frac{1}{2} \sum_{i,j} A_{ij} \left(Z_{i,r} - Z_{j,r}\right)^2 \geq \lambda_2 \sum_i d_i (Z_{i,r} - \mu_r)^2 \\
\Rightarrow & \frac{1}{2} \sum_{i,j} A_{ij} \|z_i - z_j\|_2^2 \geq \lambda_2 \sum_i d_i \|z_i - \mu\|_2^2 \\
\Rightarrow & \frac{1}{2} \sum_{i,j} \frac{A_{ij}}{\sum_k d_k} \|z_i - z_j\|_2^2 = \frac{1}{2} \sum_{i,j} \frac{A_{ij}}{\sum_{i,j} A_{ij}} \|z_i - z_j\|_2^2 \geq \lambda_2 \sum_i \frac{d_i}{\sum_k d_k} \|z_i - \mu\|_2^2
\end{aligned}
$$

If we pick i with probability $\pi_i = d_i / \sum_k d_k$ then pick j with probability $P_{ij} = A_{ij}/d_i$, the joint probability of the ordered pair (i, j) is

$$
\mathbb{P}\{(i,j)\} = \pi_i P_{ij} = \frac{d_i}{\sum_k d_k} \cdot \frac{A_{ij}}{d_i} = \frac{A_{ij}}{\sum_k d_k} = \frac{A_{ij}}{\sum_{i,j} A_{ij}}
$$

We also have

$$
\begin{aligned}
\lambda_2 \sum_i \frac{d_i}{\sum_k d_k} \|z_i - \mu\|_2^2 &= \frac{\lambda_2}{\sum_k d_k} \sum_i d_i (\|z_i\|_2^2 + \|\mu\|_2^2 - 2 z_i^\top \mu) \\
&= \lambda_2 \left(1 + \|\mu\|_2^2 - 2\mu^\top \frac{\sum_i d_i z_i}{\sum_k d_k}\right) \\
&= \lambda_2 (1 + \|\mu\|_2^2 - 2\|\mu\|_2^2) \\
&= \lambda_2 (1 - \|\mu\|_2^2),
\end{aligned}
$$

which means

$$
\frac{1}{2} \mathbb{E}_{(i,j)\sim\pi P} \left[\|z_i - z_j\|_2^2\right] \geq \lambda_2 (1 - \|\mu\|_2^2)
$$

However, since all embeddings are normalized, we know that $A_{ij} = 1$ if and only if $z_i^\top z_j \geq \theta$, or

$$
\begin{aligned}
1 - \frac{1}{2} \|z_i - z_j\|_2^2 &\geq \theta \\
\Leftrightarrow \|z_i - z_j\|_2^2 &\leq 2 - 2\theta.
\end{aligned}
$$

Hence, $\mathbb{E}_{(i,j)\sim\pi P} \left[\|z_i - z_j\|_2^2\right] = \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \;\middle|\; \|z_i - z_j\|_2^2 \leq 2 - 2\theta\right]$. Let p be the probability of $\|z_i - z_j\|_2^2 \leq 2 - 2\theta$; we have

$$
\begin{aligned}
\mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2\right] = p \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \;\middle|\; \|z_i - z_j\|_2^2 \leq 2 - 2\theta\right] + \\
(1 - p) \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \;\middle|\; \|z_i - z_j\|_2^2 > 2 - 2\theta\right]
\end{aligned}
$$

We also have

$$
\begin{aligned}
& \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \;\middle|\; \|z_i - z_j\|_2^2 \leq 2 - 2\theta\right] \leq 2 - 2\theta \\
& \leq \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \;\middle|\; \|z_i - z_j\|_2^2 > 2 - 2\theta\right],
\end{aligned}
$$

which means

$$
\mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2\right] \geq \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \;\middle|\; \|z_i - z_j\|_2^2 \leq 2 - 2\theta\right].
$$

With $\|z_i - z_j\|_2^2 \in [0, 4]$, by the convexity of e^{-tx} we have

$$
e^{-t\|z_i - z_j\|_2^2} \leq e^{-t \cdot 0} + \frac{e^{-t \cdot 4} - 1}{4 - 0} (\|z_i - z_j\|_2^2 - 0) = 1 - \frac{1 - e^{-4t}}{4} \|z_i - z_j\|_2^2.
$$

Therefore,

$$\begin{aligned}
\mathcal{L}_{\text{unif}} &= \log \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[e^{-t\|z_i - z_j\|_2^2} \right] \\
&\leq \log \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[1 - \frac{1 - e^{-4t}}{4} \|z_i - z_j\|_2^2 \right] \\
&\leq -\frac{1 - e^{-4t}}{4} \mathbb{E}_{z_i, z_j \overset{\text{iid}}{\sim} p_{\text{data}}} \left[\|z_i - z_j\|_2^2 \right] \\
&\leq -\frac{1 - e^{-4t}}{4} \mathbb{E}_{(i,j) \sim \pi P} \left[\|z_i - z_j\|_2^2 \right] \\
&\leq -\frac{1 - e^{-4t}}{2} \lambda_2 (1 - \|\mu\|_2^2),
\end{aligned}$$

which means

$$\mathbb{E}_{L^{(1)}, L^{(2)} \sim \mathbb{P}_{L|\mathcal{G}}} [\mathcal{L}_{\text{unif}}] = \mathcal{L}_{\text{unif}} \leq -\frac{1 - e^{-4t}}{2} \mathbb{E}_{L \sim \mathbb{P}_{L|\mathcal{G}}} \left[\lambda_2 (1 - \|\mu\|_2^2) \right].$$

By Cauchy–Schwarz,

$$\mathbb{E} \left[\lambda_2 (1 - \|\mu\|_2^2) \right] \geq \mathbb{E} \left[\lambda_2 \right] \mathbb{E} \left[1 - \|\mu\|_2^2 \right] - \sqrt{\text{Var}[\lambda_2] \, \text{Var} \left[1 - \|\mu\|_2^2 \right]}.$$

Since $(1 - \|\mu\|_2^2) \in [0, 1]$, $\text{Var} \left[1 - \|\mu\|_2^2 \right] \leq 1/4$. Moreover,

$$\begin{aligned}
\mathbb{E} \left[(\lambda_2 - \overline{\lambda}_2)^2 \right] &= \mathbb{E} \left[(\lambda_2 - \mathbb{E}[\lambda_2] + \mathbb{E}[\lambda_2] - \overline{\lambda}_2)^2 \right] \\
&= \mathbb{E} \left[(\lambda_2 - \mathbb{E}[\lambda_2])^2 \right] + \mathbb{E} \left[(\mathbb{E}[\lambda_2] - \overline{\lambda}_2)^2 \right] + 2\mathbb{E} \left[(\lambda_2 - \mathbb{E}[\lambda_2])(\mathbb{E}[\lambda_2] - \overline{\lambda}_2) \right] \\
&= \text{Var}[\lambda_2] + \mathbb{E} \left[(\mathbb{E}[\lambda_2] - \overline{\lambda}_2)^2 \right],
\end{aligned}$$

which means

$$\text{Var}[\lambda_2] \leq \mathbb{E} \left[(\lambda_2 - \overline{\lambda}_2)^2 \right] \leq \frac{1}{2} \mathbb{E}[\mathcal{L}_G].$$

Hence,

$$\mathcal{L}_{\text{unif}} \leq -\frac{1 - e^{-4t}}{2} \mathbb{E}_{L \sim \mathbb{P}_{L|\mathcal{G}}} \left[\lambda_2 \right] \left(1 - \mathbb{E} \left[\|\mu\|_2^2 \right] \right) + \frac{1 - e^{-4t}}{4} \sqrt{\frac{1}{2} \mathbb{E}_{L^{(1)}, L^{(2)} \sim \mathbb{P}_{L|\mathcal{G}}} [\mathcal{L}_G]}$$

We also have

$$\begin{aligned}
\mathbb{E}[\lambda_2] &= \overline{\lambda}_2 + \mathbb{E}[\lambda_2 - \overline{\lambda}_2] \\
&\geq \overline{\lambda}_2 - \mathbb{E}[|\lambda_2 - \overline{\lambda}_2|] \\
&\geq \overline{\lambda}_2 - \sqrt{\mathbb{E} \left[(\lambda_2 - \overline{\lambda}_2)^2 \right]} \quad \text{(by Cauchy-Schwarz)} \\
&\geq \overline{\lambda}_2 - \sqrt{\frac{1}{2} \mathbb{E}_{L^{(1)}, L^{(2)} \sim \mathbb{P}_{L|\mathcal{G}}} [\mathcal{L}_G]},
\end{aligned}$$

which means

$$\begin{aligned}
\mathcal{L}_{\text{unif}} &\leq -\frac{1 - e^{-4t}}{2} \left(\overline{\lambda}_2 - \sqrt{\frac{1}{2} \mathbb{E}[\mathcal{L}_G]} \right) \left(1 - \mathbb{E} \left[\|\mu\|_2^2 \right] \right) + \frac{1 - e^{-4t}}{4} \sqrt{\frac{1}{2} \mathbb{E}[\mathcal{L}_G]} \\
&= \frac{1 - e^{-4t}}{2\sqrt{2}} \left(\frac{3}{2} - \mathbb{E} \left[\|\mu\|_2^2 \right] \right) \sqrt{\mathbb{E}_{L^{(1)}, L^{(2)} \sim \mathbb{P}_{L|\mathcal{G}}} [\mathcal{L}_G]} - \frac{(1 - e^{-4t})}{2} \overline{\lambda}_2 (1 - \mathbb{E} \left[\|\mu\|_2^2 \right]).
\end{aligned}$$

C Ablation study on p

The ablation study results on different value of p is provided in Table 4.

D Ablation study on β

The ablation study results on different value of β is provided in Figure 4.

Table 4: Accuracy $(\%)$ for several value of p. The results indicate that SpecMatch-CL's performance is highly sensitive to the choice of p.

Value of p	NCI1	PROTEINS	DD	MUTAG	COLLAB	RDT-B	RDT-M5K	IMDB-B
$p = 100$	80.45±0.49	74.97±0.56	80.22±0.89	90.14±0.87	73.21±0.74	90.71±1.07	56.34±0.96	**74.75**±0.64
$p = 80$	**81.86**±0.36	**76.69**±0.50	**81.12**±1.04	**90.87**±0.75	**74.26**±0.94	91.31±0.79	**57.22**±1.20	73.35±0.44
$p = 60$	79.32±0.58	75.39±0.43	80.45±1.12	89.39±0.78	73.61±0.87	**92.48**±0.52	55.84±1.12	72.81±0.56

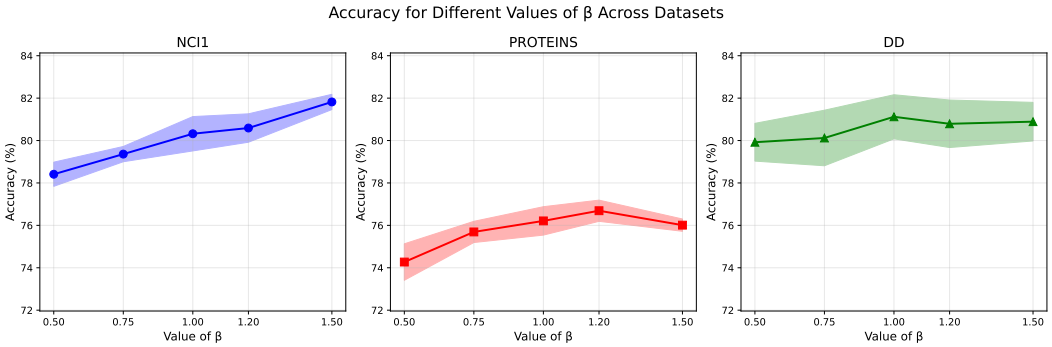

Figure 4: Ablation study for β accross NCI1, PROTEINS and DD. As shown in the plot, β has a significant effects on the performance of Specmatch-CL.

E More about experiment setting

Following GraphCL's settings, we train with Adam optimizer a learning rate selected from {0.01, 0.001, 0.0001} via grid search; we use a batch size of 512 for most dataset except MUTAG (for which we use the whole datasets as a batch); and we select the number of epochs from {20, 40, 60, 80, 100}. All experiments are run on a single NVIDIA A100 (80GB VRAM), and each experiments take about 2-4 hours.

F Dataset statistics

Dataset statistics are provided in Table 5 and 6.

Table 5: Dataset statistics for unsupervised and semi-supervised experiments.

Datasets	**Category**	**Graph Num.**	**Avg. Node**	**Avg. Degree**
NCI1	Biochemical Molecules	4110	29.87	1.08
PROTEINS	Biochemical Molecules	1113	39.06	1.86
DD	Biochemical Molecules	1178	284.32	715.66
MUTAG	Biochemical Molecules	188	17.93	19.79
COLLAB	Social Networks	5000	74.49	32.99
RDT-B	Social Networks	2000	429.63	1.15
RDB-M	Social Networks	2000	429.63	497.75
IMDB-B	Social Networks	1000	19.77	96.53

Table 6: Dataset statistics for transfer learning.

Datasets	**Category**	**Utilization**	**Graph Num.**	**Avg. Node**	**Avg. Degree**
ZINC 2M	Biochemical Molecules	Pre-Training	2,000,000	26.62	57.72
PPI-306K	Protein–Protein Interaction Nets	Pre-Training	306,925	39.82	729.62
BBBP	Biochemical Molecules	Finetuning	2,039	24.06	51.90
ToxCast	Biochemical Molecules	Finetuning	8,576	18.78	38.52
SIDER	Biochemical Molecules	Finetuning	1,427	33.64	70.71