PHeLP: Prompt-Enhanced Link Prediction for Multi-Label Node Classification on Heterophilic Graphs

Anonymous authors
Paper under double-blind review

Abstract

Heterophily complicates multi-label node classification because the same neighborhood can play different predictive roles for different candidate labels. Existing heterophily-oriented methods are largely designed for single-label classification, while conventional multi-label methods remain limited in distinguishing label-dependent neighborhood relations under heterophily. To address this challenge, we propose PHeLP, a prompt-enhanced link prediction framework for multi-label node classification on heterophilic graphs. First, to explicitly capture node–label relations, we formulate the task as heterogeneous data–label link prediction, making each candidate data–label pair an explicit prediction instance and allowing graph structure, observed training associations, and label dependencies to jointly inform prediction. Second, to distinguish the predictive roles of the same neighborhood across candidate labels, we introduce a heterophily-aware pairwise prompting mechanism. PHeLP captures affinity and repulsion as complementary neighborhood patterns and uses affinity and repulsion prompts to refine a shared pair representation in separate branches. Matching the target node's local observed-label composition to the candidate label's affinity and repulsion patterns then determines how the branch predictions are adaptively combined. Theoretical analysis indicates that PHeLP can enhance distinguishability and reduce prediction risk. Extensive experiments on multi-label graph benchmarks demonstrate the effectiveness and robustness of PHeLP across node-split and label-split settings.

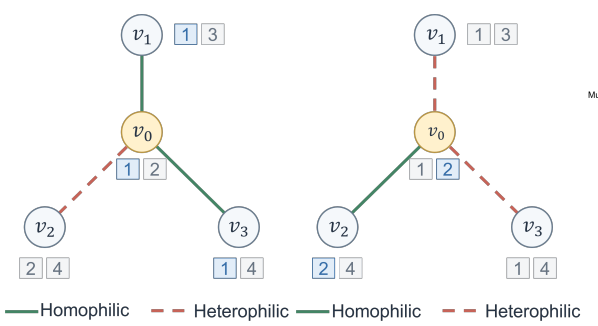
1 Introduction

Heterophily, where connected nodes tend to have different labels, challenges graph learning methods that rely on label agreement between neighbors (Zhu et al., 2020). This notion of label agreement becomes more nuanced in multi-label node classification (MLNC), where each node may carry multiple non-exclusive labels. Connected nodes can share some labels while differing on others, so a binary same-class/different-class distinction cannot fully characterize their relations (Zhao et al., 2023). The coexistence of heterophily and overlapping label assignments therefore complicates the use of graph structure for multi-label prediction.

Existing studies offer complementary perspectives on this challenge. Heterophilic graph learning improves node representations through non-local neighbor extension (Abu-El-Haija et al., 2019; Song et al., 2023; Pei et al., 2020), architecture refinement (Bo et al., 2021; Zhu et al., 2021; Liang et al., 2023; Xu et al., 2018), and hybrid designs (Zhu et al., 2020; Zheng et al., 2026). These methods have primarily been studied in single-label settings. In parallel, MLNC methods use label semantics to guide neighborhood aggregation and node representation learning (Xiao et al., 2022; Zhou et al., 2021). Label-dependency methods further exploit joint node–label representations, label-aware graph structures, or inter-label influence propagation (Shi et al., 2022; Bei et al., 2025; Sun et al., 2025), while other approaches adapt graph exploration or message passing to multi-label prediction (Akujuobi et al., 2019; Wu et al., 2026). Despite these advances, exploiting label-specific heterophilic neighborhood relations for individual node–label predictions remains underexplored.

As illustrated in Figs. 1a and 1b, changing the candidate label while holding the target node and its neighborhood fixed changes the relevant neighborhood label relations. This indicates that, in MLNC,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Relations for label 1 (b) Relations for label 2

	BlogCat	HumLoc	PCG	DBLP	EukLoc	Delve
Edge-H	10.32	40.30	17.40	75.52	42.93	51.90
Node-H	14.51	36.96	18.93	80.53	40.80	54.41
Label-H	19.05	41.81	31.34	95.19	45.09	61.06
Multi-label Ratio	27.66	17.89	45.96	15.28	18.05	18.99
LANC	67.94	59.63	56.58	91.68	51.25	66.85
LIP-GCN	70.21	73.22	64.48	94.38	72.92	94.33
CorGCN	53.99	74.33	64.53	94.49	76.01	96.90
H2GCN	72.35	71.94	60.35	95.84	73.50	98.07
GPR-GNN	72.03	71.05	61.95	95.98	74.55	98.24
LINKX	67.71	72.13	59.65	91.65	72.04	87.06
PHeLP-GCN	74.29	77.01	64.74	96.74	76.56	96.72

(c) Graph statistics and performance

Figure 1: Illustration of neighborhood relations in multi-label graphs and empirical analysis. H denotes homophily. Metric definitions are provided in Appendix A.2.2.

the same neighborhood cannot be characterized by a single notion of label agreement. In particular, neighboring labels may be systematically over- or underrepresented around nodes carrying a candidate label relative to their background frequencies, revealing label-dependent neighborhood patterns beyond direct label agreement (Ma et al., 2022). Meanwhile, the target node has its own observed neighborhood label composition, so the same global neighborhood pattern may be manifested differently across individual nodes. Consequently, the predictive role of a neighborhood depends on both the candidate label and the particular neighborhood of the target node. Fig. 1c further shows substantial variation in homophily and multi-label characteristics across datasets, highlighting the need to characterize such label-dependent neighborhood relations beyond a fixed notion of homophily.

Motivated by this perspective, we propose PHeLP, a prompt-enhanced link prediction framework for MLNC on heterophilic graphs. First, to explicitly capture the relations between data nodes and candidate labels, we reformulate MLNC as heterogeneous data–label link prediction, in which each candidate data–label pair constitutes an explicit prediction instance. By introducing labels as graph entities alongside data nodes, this formulation allows graph structure, observed data–label associations, and label dependencies to jointly inform candidate-pair predictions through learned data-node and label-node representations.

Second, we introduce heterophily-aware pairwise prompting to distinguish the predictive roles of the same neighborhood across candidate labels. To capture cross-label neighborhood relations beyond label agreement, we estimate affinity and repulsion vectors from observed training labels. These vectors characterize how frequently neighboring labels occur above or below their baseline frequencies around nodes carrying each candidate label. We group labels separately according to the similarity of their affinity and repulsion vectors and assign each group a learnable prompt. The resulting affinity and repulsion prompts refine a shared pair representation in separate branches, preserving the distinction between the two neighborhood patterns. Matching the target node's local observed-label composition to the candidate label's affinity and repulsion vectors then yields pair-specific weights for combining the two branch predictions.

Theoretical analysis indicates that PHeLP can enhance distinguishability and reduce prediction risk. Extensive experiments on multi-label graph benchmarks across multiple settings and GNN backbones demonstrate its effectiveness. Our main contributions are summarized as follows:

- We reformulate MLNC on heterophilic graphs as heterogeneous data–label link prediction, introducing labels as graph entities and treating each candidate data–label relation as a link to be predicted.
- We propose heterophily-aware pairwise prompting that matches the target node's local observed-label composition with global affinity and repulsion patterns to capture label-specific neighborhood effects.
- Theoretical analysis indicates that PHeLP can enhance distinguishability and reduce prediction risk. Extensive experiments on multi-label graph benchmarks demonstrate the effectiveness of PHeLP across node-split and label-split settings.

2 Related Work

Multi-Label Node Classification. Existing MLNC methods incorporate label information at different stages of graph learning. Label-semantic methods use label embeddings to guide neighborhood aggregation or node representation learning (Xiao et al., 2022; Zhou et al., 2021), while label-dependency methods capture correlations or influences among labels through joint node–label representations, label-aware graph structures, or influence propagation (Gao et al., 2019; Shi et al., 2022; Song et al., 2021; Bei et al., 2025; Sun et al., 2025). Other approaches adapt graph exploration or message passing itself to multi-label prediction (Akujuobi et al., 2019; Wu et al., 2026). However, exploiting label-specific heterophilic neighborhood relations for individual node–label predictions remains underexplored.

Learning on Heterophilic Graphs. Heterophily challenges conventional message-passing GNNs, as aggregating dissimilar neighbors can degrade representations. Existing methods mainly address this issue through non-local neighbor extension, architecture refinement, and hybrid designs (Zheng et al., 2026). Non-local methods expand the receptive field by mixing higher-order neighborhoods or discovering informative distant nodes (Abu-El-Haija et al., 2019; Song et al., 2023; Pei et al., 2020), while architecture-based methods distinguish information from similar and dissimilar neighbors or combine representations across multiple depths (Bo et al., 2021; Zhu et al., 2021; Liang et al., 2023; Xu et al., 2018). Hybrid designs further combine structural adaptation with specialized learning mechanisms (Zhu et al., 2020). However, these methods primarily target single-label settings, leaving the label-dependent nature of heterophily in MLNC insufficiently explored.

3 Method

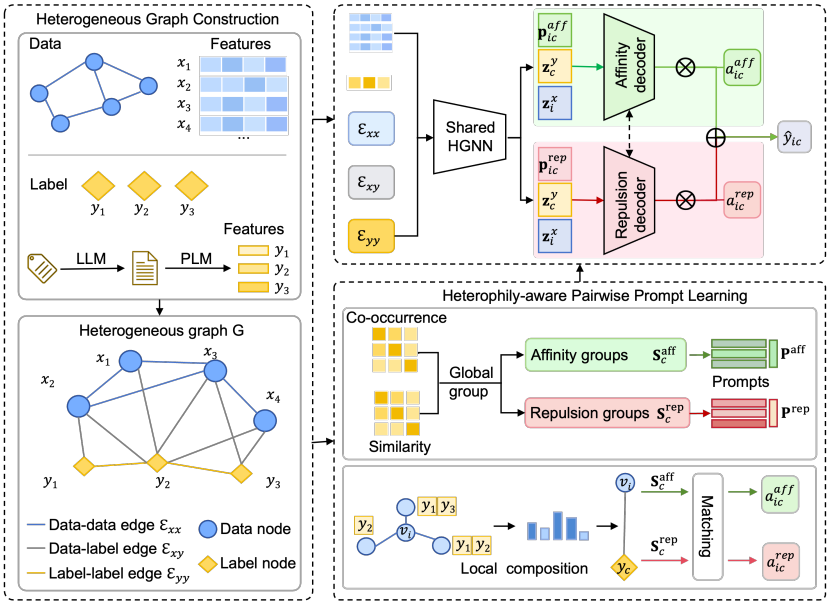


Figure 2: Overview of PHeLP. Given a multi-label graph, we first construct a heterogeneous graph over data and label nodes and employ a relation-specific encoder to jointly learn their representations. We then estimate affinity and repulsion patterns for each label and match them against the target node's neighborhood label distribution. The corresponding group prompts separately refine a shared data–label pair representation. A shared predictor produces two branch scores, which are adaptively combined using the pairwise matching weights to obtain the final association probability.

We approach heterophilic MLNC from two perspectives: data–label link prediction and label-dependent neighborhood characterization. Accordingly, PHeLP reformulates MLNC as heterogeneous data–label link prediction and further introduces heterophily-aware pairwise prompting to

capture label-dependent neighborhood patterns for each data–label pair. The overall framework is illustrated in Figure 2.

3.1 Heterogeneous Graph Construction

Given a data graph $\mathcal{G}_x = (\mathcal{V}_x, \mathcal{E}_{xx})$ with N data nodes and C candidate labels, we construct a heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$ by introducing one node for each label. The node set is $\mathcal{V} = \mathcal{V}_x \cup \mathcal{V}_y$, where $\mathcal{V}_x = \{v_i\}_{i=1}^N$ and $\mathcal{V}_y = \{v_c^y\}_{c=1}^C$ are disjoint sets of data nodes and label nodes, respectively.

Let Ω_{tr} denote the set of data–label pairs whose binary targets are available for training, and let $Y_{ic} \in \{0,1\}$ denote the target of pair (i,c). We define the observed training-label incidence matrix $\mathbf{Y}^{\mathrm{tr}} \in \{0,1\}^{N\times C}$ as $Y_{ic}^{\mathrm{tr}} = \mathbb{I}\{(i,c) \in \Omega_{\mathrm{tr}}\}Y_{ic}$, where $\mathbb{I}\{\cdot\}$ denotes the indicator function. Under the node-split protocol, $\Omega_{\mathrm{tr}} = \{(i,c) \mid v_i \in \mathcal{V}_x^{\mathrm{tr}},\ c \in \{1,\dots,C\}\}$, where $\mathcal{V}_x^{\mathrm{tr}}$ is the training node set. Under the label-split protocol, Ω_{tr} consists of the data–label pairs assigned to training.

We retain the original data–data edges $\mathcal{E}_{xx}$ and connect each data node to its observed positive labels:

$$\mathcal{E}_{xy} = \{(v_i, v_c^y) \mid Y_{ic}^{\mathrm{tr}} = 1\}, \qquad \mathcal{E}_{yx} = \{(v_c^y, v_i) \mid (v_i, v_c^y) \in \mathcal{E}_{xy}\}. \tag{1}$$

The reverse edges $\mathcal{E}_{yx}$ enable bidirectional message passing between data nodes and label nodes. The label–label edge set $\mathcal{E}_{yy} \subseteq \mathcal{V}_y \times \mathcal{V}_y$ captures inter-label relations derived from label co-occurrence, prior knowledge, or semantic similarity between label descriptions. Label co-occurrence statistics are computed from $\mathbf{Y}^{\mathrm{tr}}$.

The resulting edge set is $\mathcal{E} = \mathcal{E}_{xx} \cup \mathcal{E}_{xy} \cup \mathcal{E}_{yx} \cup \mathcal{E}_{yy}$, with relation-type set $\mathcal{R}$. This heterogeneous graph integrates data connectivity, observed data–label associations, and inter-label relations for joint representation learning.

3.2 Heterogeneous Representation Learning

We jointly encode data nodes and label nodes on the constructed heterogeneous graph to obtain their base representations for subsequent pairwise prompting and link prediction.

For data nodes, we use the original attributes provided by each dataset, denoted by $\mathbf{x}_i$. To equip label nodes with semantic features, we first use a large language model (LLM) to enrich each label description based on its name and general meaning. A pretrained language model (PLM) then encodes the enriched description:

$$\mathbf{x}_c^y = \mathrm{PLM}\left(\mathrm{LLM}(t_c)\right), \qquad c = 1,\dots,C, \tag{2}$$

where t_c denotes the textual information associated with label c. The prompt template is provided in Appendix A.2.3. We use type-specific input projections when needed to align feature dimensions, and denote the resulting initial representation of a node $v \in \mathcal{V}$ by $\mathbf{h}_v^{(0)}$.

The encoder first aggregates messages within each relation and then combines them across relations. Let $\mathcal{N}_\xi(v)$ denote the neighbors of node v connected through relation $\xi \in \mathcal{R}$. At layer ℓ, the relation-specific message is

$$\mathbf{m}_{v,\xi}^{(\ell)} = \sum_{u\in\mathcal{N}_\xi(v)} \frac{1}{\kappa_{v,\xi}} \psi\left(\mathbf{W}_\xi^{(\ell)}\mathbf{h}_u^{(\ell-1)} + \mathbf{O}^{(\ell)}\mathbf{e}_{uv}\right), \tag{3}$$

where $\mathbf{W}_\xi^{(\ell)}$ is a relation-specific transformation, $\mathbf{e}_{uv}$ is the feature of the directed edge (u,v), and $\mathbf{O}^{(\ell)}$ projects edge features into the message space. Here, $\kappa_{v,\xi} > 0$ is a normalization factor and $\psi(\cdot)$ is a nonlinear activation. An empty relation-specific neighborhood contributes a zero message.

The node representation is then updated as

$$\mathbf{h}_v^{(\ell)} = \mathbf{U}^{(\ell)}\left[\psi\left(\sum_{\xi\in\mathcal{R}} \mathbf{m}_{v,\xi}^{(\ell)} + \mathbf{W}_0^{(\ell)}\mathbf{h}_v^{(\ell-1)}\right) \| \mathbf{h}_v^{(\ell-1)}\right], \tag{4}$$

where $\mathbf{W}_0^{(\ell)}$ transforms the node's own representation, $\|$ denotes concatenation, and $\mathbf{U}^{(\ell)}$ projects the concatenated features to the next-layer representation. This update combines relation-specific neighborhood information with a skip connection to the preceding node representation.

After L layers, we obtain data-node embeddings $\mathbf{z}_i = \mathbf{h}_{v_i}^{(L)}$ and label-node embeddings $\mathbf{z}_c^y = \mathbf{h}_{v_c^y}^{(L)}$. For a given candidate label c, the encoder produces a shared representation $\mathbf{z}_c^y$ across target data nodes.

3.3 Heterophily-Aware Pairwise Prompt Learning

The same neighborhood can contribute differently to different candidate labels. We capture such label-dependent effects by combining global neighborhood label patterns with the locally observed labels of the target node.

We first estimate neighborhood label relations from the observed training labels $\mathbf{Y}^{\mathrm{tr}}$, where $\mathbf{y}_j^{\mathrm{tr}} = (Y_{j1}^{\mathrm{tr}}, \ldots, Y_{jC}^{\mathrm{tr}})^\top$, over the data–data edges in the training set $\mathcal{E}_{xx}^{\mathrm{tr}} \subseteq \mathcal{E}_{xx}$. For each edge (v_j, v_i), v_j is treated as the neighbor and v_i as the center. Let $D_c = \sum_{(v_j,v_i)\in\mathcal{E}_{xx}^{\mathrm{tr}}} Y_{ic}^{\mathrm{tr}}$ denote the number of training edges whose center has label c. For neighboring label b, we define $P_{cb}^{\mathrm{edge}} = \frac{\sum_{(v_j,v_i)\in\mathcal{E}_{xx}^{\mathrm{tr}}} Y_{ic}^{\mathrm{tr}} Y_{jb}^{\mathrm{tr}}}{\max(D_c,1)}$ as its conditional frequency around label c, and $\pi_b^{\mathrm{edge}} = \frac{\sum_{(v_j,v_i)\in\mathcal{E}_{xx}^{\mathrm{tr}}} Y_{jb}^{\mathrm{tr}}}{\max(|\mathcal{E}_{xx}^{\mathrm{tr}}|,1)}$ as its marginal neighbor-label frequency. Comparing the two removes the effect of overall label prevalence and highlights neighborhood patterns associated with label c. We define the affinity and repulsion scores as

$$S_{cb}^{\mathrm{aff}} = \mathbb{I}\{D_c > 0\}\left[P_{cb}^{\mathrm{edge}} - \pi_b^{\mathrm{edge}}\right]_+, \qquad S_{cb}^{\mathrm{rep}} = \mathbb{I}\{D_c > 0\}\left[\pi_b^{\mathrm{edge}} - P_{cb}^{\mathrm{edge}}\right]_+, \tag{5}$$

where $[x]_+ = \max(x, 0)$. The corresponding profile of label c is $\mathbf{s}_c^r = (S_{c1}^r, \ldots, S_{cC}^r)^\top$, $r \in \{\mathrm{aff}, \mathrm{rep}\}$.

These profiles capture graph-level neighborhood patterns, whereas the usefulness of each pattern depends on the local neighborhood of the target node. We therefore summarize the observed labels around v_i by a local label-frequency vector

$$\mathbf{q}_i = \frac{\sum_{v_j\in\mathcal{N}_i^{\mathrm{tr}}} \mathbf{y}_j^{\mathrm{tr}}}{\max\left(|\mathcal{N}_i^{\mathrm{tr}}|, 1\right)}, \tag{6}$$

where $\mathcal{N}_i^{\mathrm{tr}}$ contains neighbors with observed training labels.

For each candidate pair (v_i, v_c^y), we compare $\mathbf{q}_i$ with the affinity and repulsion profiles of label c:

$$\mu_{ic}^r = \langle \overline{\mathbf{q}}_i, \overline{\mathbf{s}}_c^r \rangle, \qquad \alpha_{ic}^r = \begin{cases} \dfrac{\mu_{ic}^r}{\mu_{ic}^{\mathrm{aff}} + \mu_{ic}^{\mathrm{rep}}}, & \mu_{ic}^{\mathrm{aff}} + \mu_{ic}^{\mathrm{rep}} > 0, \\ \dfrac{1}{2}, & \text{otherwise}, \end{cases} \tag{7}$$

where the overline denotes ℓ_1 normalization, with zero vectors mapped to zero. The matching weight α_{ic}^r therefore links the local neighborhood of v_i to the affinity or repulsion pattern of label c.

To share prompt parameters across labels with similar neighborhood patterns, we group their affinity and repulsion profiles separately using cosine similarity. Labels with similar affinity or repulsion patterns share a learnable group prompt, denoted by $\mathbf{p}_c^r$ for label c and $r \in \{\mathrm{aff}, \mathrm{rep}\}$.

Given the encoded node representations $\mathbf{z}_i$ and $\mathbf{z}_c^y$, we first construct the pair representation $\mathbf{h}_{ic} = \phi_{\mathrm{pair}}\left([\mathbf{z}_i \| \mathbf{z}_c^y \| \mathbf{z}_i \odot \mathbf{z}_c^y \| |\mathbf{z}_i - \mathbf{z}_c^y|]\right)$, where ϕ_{pair} is a trainable pair encoder, $\odot$ denotes element-wise multiplication, and the absolute value is applied element-wise.

To allow the same shared prompt to contribute differently to different data–label pairs, we introduce a pair-conditioned gate. For each branch $r \in \{\mathrm{aff}, \mathrm{rep}\}$, we project the selected group prompt into the pair-representation space as $\mathbf{u}_c^r = \mathbf{W}_p \mathbf{p}_c^r$. We then compute a gating vector conditioned jointly on the pair representation and the projected prompt $\mathbf{g}_{ic}^r = \sigma\left(\mathbf{W}_g[\mathbf{h}_{ic} \| \mathbf{u}_c^r] + \mathbf{b}_g\right)$, where $\mathbf{W}_p$ is the prompt projection, $\mathbf{W}_g$ and $\mathbf{b}_g$ are the gating parameters, and $\sigma(\cdot)$ denotes the sigmoid function. This gate modulates the prompt contribution element-wise before it is added to the pair representation:

$$\widetilde{\mathbf{h}}_{ic}^r = \mathbf{h}_{ic} + \rho \mathbf{g}_{ic}^r \odot \mathbf{u}_c^r, \qquad r \in \{\mathrm{aff}, \mathrm{rep}\}, \tag{8}$$

where ρ controls the overall prompt strength. Applying this update separately to the affinity and repulsion prompts yields two branch-specific pair representations for subsequent link prediction.

3.4 LINK PREDICTION

For each candidate data–label pair (v_i, v_c^y), we apply the shared predictor f_θ to the affinity- and repulsion-refined pair representations $\widetilde{\mathbf{h}}_{ic}^r$, $r \in \{\mathrm{aff}, \mathrm{rep}\}$, to obtain a scalar logit for each branch. We then aggregate the two logits using the pair-specific matching weights α_{ic}^r defined in Eq. 7 to obtain the predicted association probability:

$$\widehat{y}_{ic} = \sigma\left(\sum_{r\in\{\mathrm{aff},\mathrm{rep}\}} \alpha_{ic}^r f_\theta\left(\widetilde{\mathbf{h}}_{ic}^r\right)\right), \tag{9}$$

The pair-conditioned gates in Eq. 8 regulate the prompt contribution within each branch, whereas the matching weights α_{ic}^r control the contribution of each branch logit to the final prediction.

We optimize binary cross-entropy over the supervised training pairs Ω_{tr}, which include both observed positive and observed negative targets:

$$\mathcal{L}_{\mathrm{BCE}} = -\frac{1}{|\Omega_{\mathrm{tr}}|}\sum_{(i,c)\in\Omega_{\mathrm{tr}}} \left[Y_{ic}\log\widehat{y}_{ic} + (1-Y_{ic})\log(1-\widehat{y}_{ic})\right]. \tag{10}$$

3.5 THEORETICAL ANALYSIS

We analyze the two components of PHeLP in terms of candidate-pair distinguishability and attainable prediction risk. We first analyze the distinguishability of PHeLP.

Theorem 1. *For the L-layer message-passing architectures considered here, with $L \geq 2$, finite positive normalization factors, and an activation satisfying $\psi(0) = 0$ and $\psi(t) > 0$ for $t > 0$, there exist a data graph $\mathcal{G}_x$, two nodes v_i and v_j with isomorphic L-hop attributed rooted neighborhoods, observed data–label associations involving neither target node, and a candidate label c such that*

$$\forall\theta_{\mathrm{dir}}, \quad \Delta_{ij,c}^{\mathrm{dir}}(\theta_{\mathrm{dir}}) = 0, \qquad \exists\theta_{\mathrm{het}}, \quad \Delta_{ij,c}^{\mathrm{het}}(\theta_{\mathrm{het}}) > 0, \tag{11}$$

where $\Delta_{ij,c}^{\mathrm{dir}}(\theta_{\mathrm{dir}}) = |s_{ic}^{\mathrm{dir}}(\theta_{\mathrm{dir}}) - s_{jc}^{\mathrm{dir}}(\theta_{\mathrm{dir}})|$ and $\Delta_{ij,c}^{\mathrm{het}}(\theta_{\mathrm{het}}) = |s_{ic}^{\mathrm{het}}(\theta_{\mathrm{het}}) - s_{jc}^{\mathrm{het}}(\theta_{\mathrm{het}})|$, with s_{ic}^{dir} and s_{ic}^{het} denoting the prediction logits under the data-node-only and heterogeneous formulations, respectively. The heterogeneous formulation uses $\rho = 0$.

Theorem 1 shows that PHeLP can break data-node-only indistinguishability even without prompting. We next analyze the prediction risk of PHeLP.

Theorem 2. *For the following risk analyses, we fix the heterogeneous encoder, the pair encoder ϕ_{pair}, the label groups, and the training-derived relation profiles. For a fixed candidate label c, randomness comes from sampling v_i. Let $\mathcal{R}^*_{\mathrm{match},c}$ and $\mathcal{R}^*_{\mathrm{uniform},c}$ denote the infimum population BCE risks over the prompt module and shared predictor under the two fusion rules. $H(\cdot \mid \cdot)$ and $I(\cdot;\cdot \mid \cdot)$ denote conditional entropy and conditional mutual information, respectively. Assume independently trainable affinity and repulsion group prompts and a shared predictor class containing two-ReLU-unit networks with an affine output. Suppose $\rho > 0$, $\mathbf{h}_{ic}$ is bounded, and, for some $\tau_c \in [0,1]$, $\gamma_c > 0$, and $\varepsilon_c \in [0,1)$, $\Pr\big((2Y_{ic}-1)(\alpha_{ic}^{\mathrm{aff}} - \tau_c) \geq \gamma_c\big) \geq 1 - \varepsilon_c$. Then*

$$\mathfrak{R}^*_{\mathrm{uniform},c} - \mathfrak{R}^*_{\mathrm{match},c} \geq H(Y_{ic} \mid \mathbf{h}_{ic}) - \Psi(\varepsilon_c, \gamma_c), \tag{12}$$

where $\Psi(\varepsilon_c,\gamma_c) = \varepsilon_c\left(1 + \log 2 + \gamma_c^{-1}\log(1/\varepsilon_c)\right)$ for $\varepsilon_c > 0$, and $\Psi(0,\gamma_c) = 0$. The advantage is strict whenever $\Psi(\varepsilon_c,\gamma_c) < H(Y_{ic} \mid \mathbf{h}_{ic})$.

Theorem 2 shows that PHeLP with pairwise prompting yields lower attainable prediction risk than uniform fusion when the margin-based bound is below $H(Y_{ic} \mid \mathbf{h}_{ic})$. Finally, we analyze the prediction risk of our complete PHeLP.

Theorem 3. *Let X_c denote the data-node-only context, such that every direct predictor under comparison is a function of X_c, and let $\mathfrak{R}^*_{\mathrm{dir},c}$ denote the corresponding infimum population BCE risk. Under the conditions of Theorem 2, define $I_c = I(Y_{ic}; \mathbf{h}_{ic} \mid X_c)$ and $\delta_c = H(Y_{ic} \mid X_c, \mathbf{h}_{ic}) - \Psi(\varepsilon_c,\gamma_c)$. If $I_c > 0$ and $\delta_c > 0$, then*

$$\mathfrak{R}^*_{\mathrm{dir},c} - \mathfrak{R}^*_{\mathrm{match},c} \geq I_c + \delta_c > 0, \qquad \mathfrak{R}^*_{\mathrm{uniform},c} - \mathfrak{R}^*_{\mathrm{match},c} \geq \delta_c > 0. \tag{13}$$

Theorem 3 shows that under the stated conditions, PHeLP has strictly lower infimum prediction risk than both data-node-only prediction and uniform fusion. All proofs are provided in Appendix A.1.

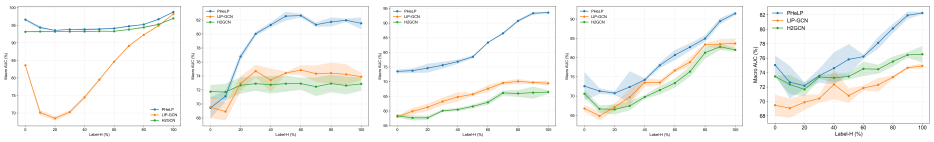
4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

We conduct all experiments on three NVIDIA GeForce RTX 3090 GPUs and one NVIDIA GeForce RTX 4090 GPU. We compare PHeLP against three groups of baselines: (1) a conventional GNN-based multi-label baseline (Kipf & Welling, 2017; Wang et al., 2023); (2) graph-based MLNC methods (Akujuobi et al., 2019; Gao et al., 2019; Xiao et al., 2022; Zhou et al., 2021; Song et al., 2021; Bei et al., 2025; Sun et al., 2025); and (3) heterophily-oriented GNNs (Zhu et al., 2020; Chien et al., 2021; Bo et al., 2021; Lim et al., 2021). For the third group, we replace the original single-label prediction heads with multi-label sigmoid classifiers and adopt the same node-split protocol for fair comparison. We evaluate our method on six public multi-label graph datasets, with dataset statistics provided in Appendix A.2.6. We use GPT-5.5 to generate label descriptions and encode them with domain-specific pretrained language models: SciBERT (Beltagy et al., 2019) for DBLP, and PubMedBERT (Gu et al., 2021) for HumLoc and EukLoc. The resulting semantic representations can be used for label-node features or semantic label–label relations. For BlogCat, PCG, and Delve, where suitable textual label descriptions are unavailable, we instead use one-hot label features. We follow the dataset splits of Sun et al. (2025).

4.2 EXPERIMENTAL RESULTS

Case Study. To evaluate PHeLP under varying label homophily, we adjust Label-H defined in Eq. 74 via degree-preserving edge rewiring. As shown in Figure 3, all methods are affected by changes in Label-H, while PHeLP outperforms the baselines in most settings and maintains its advantage at lower Label-H. These results support our design of adapting neighborhood information for each data–label pair, which enables PHeLP to better capture label-dependent neighborhood patterns under heterophily.



(a) DBLP (b) BlogCat (c) PCG (d) HumLoc (e) EukLoc

Figure 3: Case-study results under varying Label-H levels.

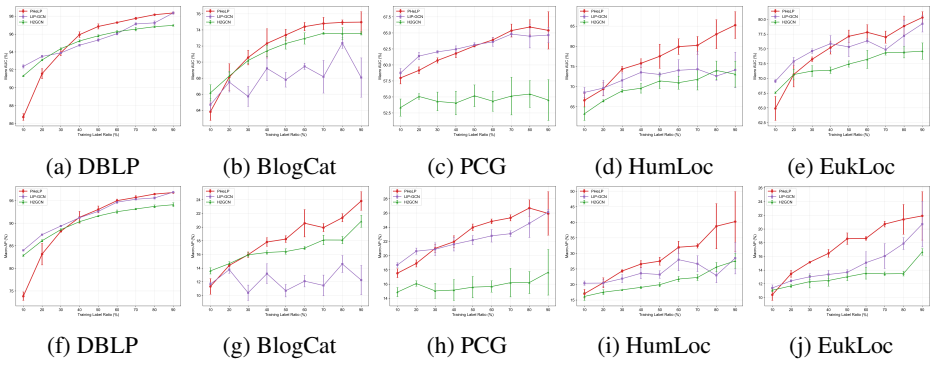
Results on Node Classification. We compare PHeLP with multi-label classification methods and heterophily-oriented GNNs under node-split and label-split settings. Tables 1 and 2 report the results. Under the node split, PHeLP consistently delivers strong performance across datasets and encoders. Applying PHeLP to heterophily-oriented backbones generally improves their performance, indicating that the improvements are not tied to a specific graph encoder. Under the label split, PHeLP also achieves competitive or superior results across datasets, demonstrating its effectiveness in predicting unobserved data–label associations from partially observed supervision.

Results under Label Splitting with Varying Training Ratios. Under the label-split protocol, we vary the training ratio from 10% to 90% to evaluate performance. As shown in Fig. 4, PHeLP becomes increasingly competitive as more training labels are observed, with clear gains on HumLoc and EukLoc and stronger performance at higher ratios on PCG. On BlogCat, the advantage is more evident in Macro-AP, while on DBLP the gap with the strongest baseline gradually narrows. These results demonstrate that PHeLP benefits from richer observed label information for learning label-dependent relations.

Ablation Study. We compare three variants: w/o Prompt removes pairwise prompting; w/o LP additionally replaces data–label link prediction with a direct MLNC head; and Uniform retains both prompted branches with weights fixed to $1/2$. All variants are trained independently under the same protocol. Figure 5a shows that removing prompting consistently degrades performance, while

Table 1: Performance under node split setting with the split of 6:2:2 (mean ± std. deviation) in terms of Macro AUC and Macro AP. The top performance are **bolded** and second ones are underlined.

Method	DBLP		BlogCat		Delve	
	Macro AUC	Macro AP	Macro AUC	Macro AP	Macro AUC	Macro AP
GCN-PLAIN	80.55 ± 1.23	73.44 ± 1.12	63.95 ± 1.85	10.21 ± 0.56	51.05 ± 2.13	0.93 ± 0.12
MLGW	73.32 ± 1.44	56.03 ± 0.47	60.02 ± 2.19	9.81 ± 0.46	97.98 ± 0.23	75.99 ± 0.79
ML-GCN	72.66 ± 2.73	56.71 ± 2.59	60.97 ± 2.21	9.68 ± 0.39	84.28 ± 0.18	3.78 ± 0.01
LARN	74.29 ± 2.53	58.11 ± 1.25	63.18 ± 1.84	9.77 ± 0.63	$\mathbf{98.76 \pm 0.03}$	$\mathbf{84.33 \pm 0.24}$
LANC	91.68 ± 0.42	83.50 ± 0.83	67.94 ± 3.30	10.35 ± 0.82	66.85 ± 7.03	11.23 ± 4.19
VariMul	92.14 ± 1.23	85.30 ± 0.92	68.71 ± 2.97	13.74 ± 0.83	82.95 ± 0.59	34.80 ± 1.13
CorGCN	94.49 ± 0.24	87.73 ± 0.45	53.99 ± 0.43	4.46 ± 0.12	96.90 ± 0.05	72.80 ± 0.30
LIP-GCN	94.38 ± 1.51	87.45 ± 1.28	70.21 ± 2.02	12.33 ± 0.91	94.33 ± 0.04	23.06 ± 0.27
H2GCN	95.84 ± 0.16	91.62 ± 0.35	72.35 ± 0.62	17.14 ± 0.10	98.07 ± 0.03	28.24 ± 0.35
GPR-GNN	95.98 ± 0.13	91.54 ± 0.25	72.03 ± 0.71	17.79 ± 0.34	98.24 ± 0.04	29.74 ± 0.73
FAGCN	95.15 ± 0.58	89.71 ± 1.42	72.58 ± 2.19	17.19 ± 0.53	94.74 ± 0.09	16.47 ± 0.06
LINKX	91.65 ± 0.41	85.88 ± 0.37	67.71 ± 1.09	16.66 ± 0.37	87.06 ± 1.66	15.01 ± 0.27
PHeLP-GraphConv	96.74 ± 0.17	93.43 ± 0.44	74.29 ± 0.47	18.71 ± 0.69	96.72 ± 0.27	73.22 ± 0.37
PHeLP-H2GCN	97.38 ± 0.12	94.52 ± 0.32	74.31 ± 0.45	19.26 ± 0.38	98.16 ± 0.20	74.98 ± 1.09
PHeLP-GPRGNN	$\mathbf{97.44 \pm 0.07}$	$\mathbf{94.70 \pm 0.25}$	$\mathbf{74.80 \pm 0.78}$	18.41 ± 0.90	98.49 ± 0.15	71.45 ± 0.50
PHeLP-FAGCN	97.14 ± 0.13	93.70 ± 0.34	73.61 ± 0.64	$\mathbf{19.37 \pm 0.65}$	97.75 ± 0.36	71.45 ± 2.95
Method	PCG		HumLoc		EukLoc	
	Macro AUC	Macro AP	Macro AUC	Macro AP	Macro AUC	Macro AP
GCN-PLAIN	59.75 ± 1.70	20.16 ± 1.00	60.15 ± 1.28	14.71 ± 1.51	68.02 ± 1.10	11.72 ± 0.69
MLGW	55.86 ± 3.92	15.59 ± 1.33	56.92 ± 1.03	10.54 ± 0.43	51.77 ± 1.41	6.02 ± 0.55
ML-GCN	57.24 ± 2.44	19.45 ± 1.13	60.79 ± 1.38	14.55 ± 0.46	54.17 ± 1.88	6.92 ± 0.49
LARN	57.79 ± 0.61	19.09 ± 0.38	61.48 ± 1.22	17.89 ± 0.62	57.79 ± 1.28	7.41 ± 0.80
LANC	56.58 ± 0.63	19.51 ± 0.93	59.63 ± 1.21	13.26 ± 0.98	51.25 ± 0.63	6.11 ± 0.89
VariMul	62.77 ± 0.34	21.93 ± 0.52	67.42 ± 2.44	23.34 ± 0.88	71.44 ± 1.66	13.74 ± 0.76
CorGCN	64.53 ± 1.08	$\mathbf{24.54 \pm 0.84}$	74.33 ± 0.00	25.33 ± 0.00	76.01 ± 1.38	16.75 ± 0.54
LIP-GCN	64.48 ± 1.06	22.89 ± 0.81	73.22 ± 1.76	25.18 ± 1.16	72.92 ± 1.82	15.26 ± 0.56
H2GCN	60.35 ± 0.62	20.11 ± 0.43	71.94 ± 0.99	22.78 ± 0.62	73.50 ± 1.46	12.63 ± 0.84
GPR-GNN	61.95 ± 0.64	20.91 ± 0.35	71.05 ± 1.12	22.02 ± 1.25	74.55 ± 0.93	12.93 ± 0.22
FAGCN	55.19 ± 0.28	16.87 ± 0.31	68.46 ± 0.55	18.07 ± 0.43	74.30 ± 1.16	12.77 ± 0.12
LINKX	59.65 ± 0.56	22.21 ± 0.86	72.13 ± 2.44	23.19 ± 0.75	72.04 ± 1.75	13.37 ± 0.49
PHeLP-GraphConv	64.74 ± 0.87	24.12 ± 0.70	$\mathbf{77.01 \pm 0.72}$	$\mathbf{28.60 \pm 0.94}$	76.56 ± 0.25	15.90 ± 0.22
PHeLP-H2GCN	64.54 ± 0.92	23.57 ± 0.46	76.83 ± 0.31	25.91 ± 1.28	$\mathbf{79.55 \pm 0.58}$	$\mathbf{18.07 \pm 0.54}$
PHeLP-GPRGNN	$\mathbf{64.80 \pm 1.41}$	24.42 ± 0.58	76.32 ± 0.70	27.87 ± 1.57	78.29 ± 0.94	17.90 ± 1.59
PHeLP-FAGCN	63.98 ± 2.12	22.87 ± 2.00	76.04 ± 1.23	28.02 ± 1.10	78.84 ± 1.05	16.13 ± 0.59



(a) DBLP (b) BlogCat (c) PCG (d) HumLoc (e) EukLoc

(f) DBLP (g) BlogCat (h) PCG (i) HumLoc (j) EukLoc

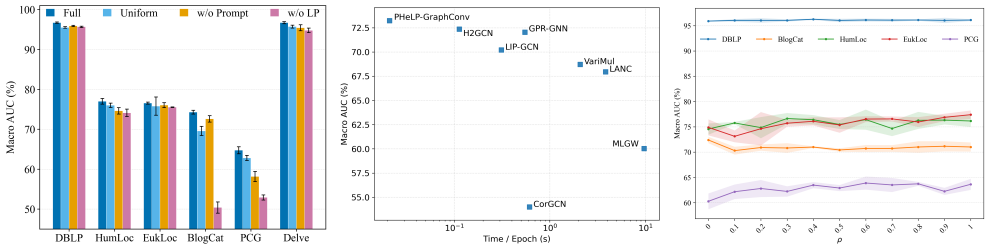
Figure 4: Performance under label splitting with varying training ratios. The top and bottom rows report Macro-AUC and Macro-AP, respectively.

replacing link prediction leads to a further decline. Full also consistently outperforms Uniform, supporting the benefit of pair-specific matching.

Efficiency Analysis. As analyzed in Appendix A.2.8, for M scored candidate pairs, the GraphConv-based PHeLP costs $O\left(L(|\mathcal{E}|d + (N+C)d^2) + M(C+d^2)\right)$, where d is the feature

Table 2: Performance under label split setting with the split of 6:2:2 in terms of Macro AUC and Macro AP.

Methods	DBLP		BlogCat		Delve	
	Macro AUC	Macro AP	Macro AUC	Macro AP	Macro AUC	Macro AP
CorGCN	96.89 ± 0.11	94.99 ± 0.35	53.16 ± 0.72	4.40 ± 0.10	96.55 ± 0.08	22.73 ± 0.46
LIP-GCN	96.04 ± 0.14	94.65 ± 0.28	69.46 ± 0.39	12.09 ± 0.84	94.30 ± 0.13	22.62 ± 0.50
H2GCN	96.27 ± 0.13	92.54 ± 0.38	72.94 ± 0.70	16.93 ± 0.23	98.45 ± 0.05	28.19 ± 0.51
GPR-GNN	96.96 ± 0.07	93.20 ± 0.15	74.20 ± 0.69	17.86 ± 0.32	98.54 ± 0.04	$\mathbf{30.53 \pm 0.94}$
PHeLP-GraphConv	97.29 ± 0.04	94.99 ± 0.18	74.41 ± 0.53	$\mathbf{20.56 \pm 2.00}$	96.63 ± 0.27	23.48 ± 4.00
PHeLP-H2GCN	97.62 ± 0.30	95.37 ± 0.50	74.00 ± 0.98	18.71 ± 0.70	98.48 ± 0.15	28.20 ± 0.87
PHeLP-GPRGNN	$\mathbf{97.68 \pm 0.21}$	$\mathbf{95.48 \pm 0.38}$	$\mathbf{75.16 \pm 0.42}$	19.33 ± 1.58	$\mathbf{98.68 \pm 0.12}$	29.46 ± 2.11
Methods	PCG		HumLoc		EukLoc	
	Macro AUC	Macro AP	Macro AUC	Macro AP	Macro AUC	Macro AP
CorGCN	62.99 ± 0.94	23.67 ± 1.02	73.83 ± 1.19	27.07 ± 1.71	75.09 ± 1.29	17.26 ± 1.11
LIP-GCN	63.60 ± 0.68	22.79 ± 0.88	74.04 ± 2.63	27.97 ± 3.58	76.36 ± 0.38	15.08 ± 1.58
H2GCN	54.33 ± 1.56	15.66 ± 0.80	70.97 ± 1.62	21.75 ± 0.83	73.22 ± 1.54	13.54 ± 0.85
GPR-GNN	57.71 ± 3.32	17.61 ± 2.42	71.23 ± 1.58	19.67 ± 1.05	73.90 ± 2.29	14.22 ± 0.22
PHeLP-GraphConv	$\mathbf{63.91 \pm 0.32}$	$\mathbf{24.82 \pm 0.34}$	$\mathbf{79.90 \pm 1.95}$	$\mathbf{31.98 \pm 1.96}$	$\mathbf{77.81 \pm 1.29}$	18.63 ± 0.29
PHeLP-H2GCN	61.91 ± 0.75	22.02 ± 0.26	76.16 ± 2.63	24.57 ± 1.52	77.74 ± 0.25	18.32 ± 0.80
PHeLP-GPRGNN	63.34 ± 0.88	23.14 ± 0.61	77.87 ± 1.41	26.74 ± 1.04	77.04 ± 1.04	$\mathbf{19.32 \pm 0.59}$



(a) Ablation results (b) Efficiency on BlogCat (c) Sensitivity to ρ

Figure 5: Ablation, efficiency analysis, and parameter sensitivity.

dimension. During training, $M = |\mathcal{V}_x^{\text{tr}}|C$ under node splitting and $M = |\Omega_{\text{tr}}|$ under label splitting. Fig. 5b shows the runtime–performance trade-off on BlogCat. PHeLP-GraphConv achieves the highest Macro-AUC and the shortest per-epoch training time, demonstrating a favorable runtime–performance trade-off.

Parameter Sensitivity Analysis. We study the sensitivity of PHeLP to the prompt strength ρ. As shown in Figs. 5c and 7b, PHeLP is generally stable over a broad range of ρ, with DBLP and HumLoc showing only small variations. EukLoc and PCG benefit from stronger prompting, whereas BlogCat prefers a smaller ρ. These results demonstrate that the optimal prompt strength is dataset-dependent, while PHeLP remains robust across a reasonably broad range of values.

5 Conclusion

We study multi-label node classification on heterophilic graphs, where the same neighborhood can play different predictive roles for different candidate labels. We propose PHeLP, a prompt-enhanced link prediction framework that reformulates this task as heterogeneous data–label link prediction, enabling prediction at the individual data–label pair level. Building on this formulation, PHeLP captures affinity and repulsion as complementary neighborhood patterns. Their group prompts separately refine a shared pair representation, while pair-specific matching weights adaptively combine the resulting predictions. Theoretical analysis indicates that PHeLP can enhance distinguishability and reduce prediction risk. Extensive experiments on multi-label graph benchmarks demonstrate the effectiveness of PHeLP across node-split and label-split settings.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish the writing of the manuscript and to assist in generating synthetic data, specifically to generate semantic label descriptions used as model inputs. All AI-assisted text and generated label descriptions were carefully reviewed and verified by the authors. We take responsibility for the final content of this work, including text, claims, experimental results, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we describe the model architecture, heterogeneous graph construction, pairwise prompting mechanism, and training objective in detail in the main paper. The appendix provides the complete theoretical assumptions and proofs, along with dataset statistics, data splits, evaluation protocols, implementation details, and dataset-specific hyperparameter configurations. All experiments are run with multiple random seeds, and we report the mean and standard deviation of the results. The source code and data needed to reproduce the experiments are available in the supplementary materials and at `https://anonymous.4open.science/r/PPPP-7D8B`.

REFERENCES

Sami Abu-El-Haija, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *Proc. of ICML*, pp. 21–29, 2019.

Uchenna Akujuobi, Han Yufei, Qiannan Zhang, and Xiangliang Zhang. Collaborative graph walk for semi-supervised multi-label node classification. In *Proc. of ICDM*, pp. 1–10, 2019. doi: 10.1109/ICDM.2019.00010.

Pablo Barcelo, Mikhail Galkin, Christopher Morris, and Miguel Romero Orth. Weisfeiler and leman go relational. In *Proc. of LoG*, volume 198, pp. 46:1–46:26, 2022.

Yuanchen Bei, Weizhi Chen, Hao Chen, Sheng Zhou, Carl Yang, Jiapei Fan, Longtao Huang, and Jiajun Bu. Correlation-aware graph convolutional networks for multi-label node classification. In *Proc. of SIGKDD*, pp. 37–48, 2025. doi: 10.1145/3690624.3709197. URL `https://doi.org/10.1145/3690624.3709197`.

Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A pretrained language model for scientific text. In *Proc. of the EMNLP*, pp. 3615–3620, 2019.

Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *Proc. of AAAI*, volume 35, pp. 3950–3957, 2021.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. In *Proc. of ICLR*, 2021.

Kaisheng Gao, Jing Zhang, and Cangqi Zhou. Semi-supervised graph embedding for multi-label graph node classification. In *Proc. of WISE*, pp. 555–567. Springer International Publishing, 2019.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM HEALTH*, 3(1), 2021.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proc. of ICLR*, 2017.

Langzhang Liang, Xiangjing Hu, Zenglin Xu, Zixing Song, and Irwin King. Predicting global label relationship matrix for graph neural networks under heterophily. In *Proc. of NeurIPS*, pp. 10909–10921, 2023.

Derek Lim, Felix Hohne, Xiuyu Li, Sijia Linda Huang, Vaishnavi Gupta, Omkar Bhalerao, and Ser Nam Lim. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Proc. of NeurIPS*, 34:20887–20902, 2021.

Yao Ma, Xiaorui Liu, Neil Shah, and Jiliang Tang. Is homophily a necessity for graph neural networks? In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ucASPPD9GKN.

Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proc. of AAAI*, volume 33, pp. 4602–4609, 2019.

Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. In *Proc. of ICLR*, 2020.

Min Shi, Yufei Tang, Xingquan Zhu, and Jianxun Liu. Multi-label graph convolutional network representation learning. *IEEE TBD*, 8(5):1169–1181, 2022.

Yunchong Song, Chenghu Zhou, Xinbing Wang, and Zhouhan Lin. Ordered gnn: Ordering message passing to deal with heterophily and over-smoothing. In *Proc. of ICLR*, 2023.

Zixing Song, Ziqiao Meng, Yifei Zhang, and Irwin King. Semi-supervised multi-label learning for graph-structured data. In *Proc. of CIKM*, pp. 1723–1733, 2021. doi: 10.1145/3459637.3482391. URL https://doi.org/10.1145/3459637.3482391.

Yifei Sun, Zemin Liu, Bryan Hooi, Yang Yang, Rizal Fathony, Jia Chen, and Bingsheng He. Multi-label node classification with label influence propagation. In *Proc. of ICLR*, 2025. URL https://openreview.net/forum?id=3X3LuwzZrl.

Lei Tang and Huan Liu. Relational learning via latent social dimensions. In *Proc. of SIGKDD*, pp. 817–826, 2009.

Haobo Wang, Shisong Yang, Gengyu Lyu, Weiwei Liu, Tianlei Hu, Ke Chen, Songhe Feng, and Gang Chen. Deep partial multi-label learning with graph disambiguation. In *Proc. of IJCAI*, 2023.

Chaokai Wu, Haofu Shi, Ningxuan Ma, Jianghong Ma, and Xiaofeng Zhang. Mlgib: Multi-label graph information bottleneck for expressive and robust message passing, 2026. URL https://arxiv.org/abs/2605.13126.

Lin Xiao, Pengyu Xu, Liping Jing, Uchenna Akujuobi, and Xiangliang Zhang. Semantic guide for semi-supervised few-shot multi-label node classification. *Information Sciences*, 591:235–250, 2022. doi: https://doi.org/10.1016/j.ins.2021.12.130. URL https://www.sciencedirect.com/science/article/pii/S0020025522000111.

Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *Proc. of ICML*, pp. 5453–5462, 2018.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *Proc. of ICLR*, 2019.

Tianqi Zhao, Thi Ngan Dong, Alan Hanjalic, and Megha Khosla. Multi-label node classification on graph-structured data. *TMLR*, 2023. URL https://openreview.net/forum?id=EZhkV2BjDP.

Xin Zheng, Yi Wang, Yixin Liu, Ming Li, Miao Zhang, Di Jin, Philip S. Yu, and Shirui Pan. Graph Neural Networks for Graphs With Heterophily: A Survey . *IEEE TKDE*, 38(07):4385–4404, 2026. doi: 10.1109/TKDE.2026.3680353.

Cangqi Zhou, Hui Chen, Jing Zhang, Qianmu Li, Dianming Hu, and Victor S. Sheng. Multi-label graph node classification with label attentive neighborhood convolution. *Expert Systems with Applications*, 180:115063, 2021. doi: https://doi.org/10.1016/j.eswa.2021.115063. URL https://www.sciencedirect.com/science/article/pii/S0957417421005042.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. In *Proc. of NeurIPS*, pp. 7793–7804, 2020.

Jiong Zhu, Ryan A Rossi, Anup Rao, Tung Mai, Nedim Lipka, Nesreen K Ahmed, and Danai Koutra. Graph neural networks with heterophily. In *Proc. of AAAI*, pp. 11168–11176, 2021.

A Appendix

A.1 Proof

A.1.1 Proof of Theorem 1

Proof. We prove the result in two steps. We first show that data-node-only message passing cannot distinguish two data nodes with isomorphic L-hop attributed rooted neighborhoods. We then construct a heterogeneous data–label graph in which the observed data–label relations break this indistinguishability.

1) Indistinguishability under data-node-only message passing. Fix an arbitrary valid parameterization θ_{dir} of an L-layer data-node-only message-passing model on $\mathcal{G}_x = (\mathcal{V}_x, \mathcal{E}_{xx})$. A generic message-passing layer can be written as

$$\mathbf{h}_v^{(\ell)} = \Phi^{(\ell)}\left(\mathbf{h}_v^{(\ell-1)}, \mathrm{AGG}^{(\ell)}\left(\left\{\mathbf{h}_u^{(\ell-1)} : u \in \mathcal{N}_x(v)\right\}\right)\right), \tag{14}$$

where $\mathcal{N}_x(v) = \{u : (u, v) \in \mathcal{E}_{xx}\}$, $\mathrm{AGG}^{(\ell)}$ is permutation invariant, and $\Phi^{(\ell)}$ and $\mathrm{AGG}^{(\ell)}$ are shared across nodes.

Let v_i and v_j have isomorphic L-hop attributed rooted neighborhoods, and let π be an attribute-preserving rooted isomorphism between them with $\pi(v_i) = v_j$. We show by induction that, for every layer $\ell = 0, \ldots, L$ and every node u satisfying $\mathrm{dist}(u, v_i) \le L - \ell$,

$$\mathbf{h}_u^{(\ell)} = \mathbf{h}_{\pi(u)}^{(\ell)}. \tag{15}$$

For $\ell = 0$, Eq. 15 follows directly from the attribute-preserving isomorphism, since corresponding nodes have identical initial features. Suppose that Eq. 15 holds at layer $\ell - 1$. For any u with $\mathrm{dist}(u, v_i) \le L - \ell$, every neighbor $w \in \mathcal{N}_x(u)$ satisfies $\mathrm{dist}(w, v_i) \le L - (\ell - 1)$. Hence, by the induction hypothesis, $\mathbf{h}_w^{(\ell-1)} = \mathbf{h}_{\pi(w)}^{(\ell-1)}$. Since π preserves adjacency, it induces a one-to-one correspondence between the neighbors of u and those of $\pi(u)$. Therefore, the two multisets of incoming representations are identical:

$$\left\{\left\{\mathbf{h}_w^{(\ell-1)} : w \in \mathcal{N}_x(u)\right\}\right\} = \left\{\left\{\mathbf{h}_{w'}^{(\ell-1)} : w' \in \mathcal{N}_x(\pi(u))\right\}\right\}. \tag{16}$$

Permutation invariance of $\mathrm{AGG}^{(\ell)}$ and the shared update function $\Phi^{(\ell)}$ then imply $\mathbf{h}_u^{(\ell)} = \mathbf{h}_{\pi(u)}^{(\ell)}$. Thus, Eq. 15 holds for all $\ell = 0, \ldots, L$.

At $\ell = L$, the condition $\mathrm{dist}(u, v_i) \le 0$ leaves only the root node, yielding

$$\mathbf{z}_i^{\mathrm{dir}} = \mathbf{h}_{v_i}^{(L)} = \mathbf{h}_{v_j}^{(L)} = \mathbf{z}_j^{\mathrm{dir}}. \tag{17}$$

This is consistent with the standard neighborhood-indistinguishability property of message-passing GNNs and their connection to the 1-WL test (Xu et al., 2019; Morris et al., 2019).

For the same candidate label c, the node-level prediction head receives the same node representation for both instances and therefore produces

$$s_{ic}^{\mathrm{dir}}(\theta_{\mathrm{dir}}) = s_{jc}^{\mathrm{dir}}(\theta_{\mathrm{dir}}). \tag{18}$$

Since θ_{dir} was chosen arbitrarily, it follows that

$$\forall \theta_{\mathrm{dir}}, \qquad \Delta_{ij,c}^{\mathrm{dir}}(\theta_{\mathrm{dir}}) = \left|s_{ic}^{\mathrm{dir}}(\theta_{\mathrm{dir}}) - s_{jc}^{\mathrm{dir}}(\theta_{\mathrm{dir}})\right| = 0. \tag{19}$$

2) Distinguishability through data–label relations. We next construct two nodes that are indistinguishable in the original data graph but become distinguishable after introducing the observed data–label relations. Consider four data nodes $\{v_i, u_i, v_j, u_j\}$ connected by two disjoint undirected edges $\{v_i, u_i\}$ and $\{v_j, u_j\}$, each represented in both directions. Let $\mathbf{x}_{v_i} = \mathbf{x}_{v_j}$ and $\mathbf{x}_{u_i} = \mathbf{x}_{u_j}$, with identical corresponding edge attributes if used. Their attributed rooted neighborhoods are therefore isomorphic at every depth.

For a candidate label c, let the only observed positive data–label association be (u_i, v_c^y), so that $Y^{\mathrm{tr}}_{u_i c} = 1$ and $Y^{\mathrm{tr}}_{u_j c} = 0$. Include its reverse edge and no label–label edges in this construction. This association concerns only the neighboring training node u_i and does not use the target annotations of v_i or v_j.

We now construct a valid parameterization under the stated encoder conditions. For this construction, take a nonzero initial representation of the label node v_c^y. Choose the first-layer label-to-data transformation to produce a positive value in an output coordinate k. Set the other relation transformations, edge-feature contributions, and self-transformation to zero, and choose the output projection to retain this coordinate from the aggregated-message block while discarding the skip-input block. Under the stated encoder conditions, the first layer yields

$$\left[\mathbf{h}^{(1)}_{u_i}\right]_k > 0, \qquad \left[\mathbf{h}^{(1)}_{u_j}\right]_k = 0. \tag{20}$$

At the second layer, choose the data-to-data transformation to transfer coordinate k with a positive coefficient. Set the remaining message and self contributions to zero, and let the output projection retain this coordinate from the aggregated-message block while discarding the skip-input block. Since each target node has exactly one data neighbor, $[\mathbf{h}^{(2)}_{v_i}]_k > 0$ and $[\mathbf{h}^{(2)}_{v_j}]_k = 0$. For $L > 2$, choose each subsequent output projection to retain coordinate k from the concatenated skip-input block and discard the aggregated-message block. Thus, for every $L \geq 2$,

$$\left[\mathbf{z}^{\mathrm{het}}_i\right]_k > 0, \quad \left[\mathbf{z}^{\mathrm{het}}_j\right]_k = 0, \quad \mathbf{z}^{\mathrm{het}}_i \neq \mathbf{z}^{\mathrm{het}}_j. \tag{21}$$

Thus, the observed data–label association creates a typed relational difference that is absent from the original data graph and can be preserved by relation-aware message passing, consistent with the expressive role of relation types in multi-relational GNNs (Barcelo et al., 2022).

It remains to show that this representation difference can be reflected in the data–label prediction score. To isolate the effect of the heterogeneous data–label formulation from pairwise prompting, we set the prompt strength to $\rho = 0$. For the same candidate label c, the two raw pair features are

$$\begin{aligned} \mathbf{x}^p_{ic} &= [\mathbf{z}^{\mathrm{het}}_i \| \mathbf{z}^y_c \| \mathbf{z}^{\mathrm{het}}_i \odot \mathbf{z}^y_c \| |\mathbf{z}^{\mathrm{het}}_i - \mathbf{z}^y_c|], \\ \mathbf{x}^p_{jc} &= [\mathbf{z}^{\mathrm{het}}_j \| \mathbf{z}^y_c \| \mathbf{z}^{\mathrm{het}}_j \odot \mathbf{z}^y_c \| |\mathbf{z}^{\mathrm{het}}_j - \mathbf{z}^y_c|]. \end{aligned} \tag{22}$$

Since their first blocks are different by Eq. 21, $\mathbf{x}^p_{ic} \neq \mathbf{x}^p_{jc}$. The pair encoder and shared predictor can be parameterized to retain and select the distinguishing coordinate, respectively, yielding different logits.

When $\rho = 0$, the two branches receive the same pair representation. Since their matching weights sum to one, their weighted combination equals the shared predictor's logit for that representation. Hence, for this valid heterogeneous parameterization θ_{het},

$$\Delta^{\mathrm{het}}_{ij,c}(\theta_{\mathrm{het}}) = \left| s^{\mathrm{het}}_{ic}(\theta_{\mathrm{het}}) - s^{\mathrm{het}}_{jc}(\theta_{\mathrm{het}}) \right| > 0. \tag{23}$$

Together with Eq. 19, this establishes the existence of the strict separation in Theorem 1.

□

BCE-risk consequence.

Corollary 4. *Under the construction of Theorem 1, consider a distribution supported on the two candidate pairs, with equal probabilities and opposite binary targets. Let $\mathcal{R}_{\mathrm{dir}}$ and $\mathcal{R}_{\mathrm{het}}$ denote the expected unweighted BCE risks on this distribution. Then*

$$\inf_{\theta_{\mathrm{dir}}} \mathcal{R}_{\mathrm{dir}} = \log 2, \qquad \inf_{\theta_{\mathrm{het}}} \mathcal{R}_{\mathrm{het}} = 0. \tag{24}$$

Proof. By Theorem 1, the two candidate pairs are indistinguishable to any data-node-only predictor and therefore receive the same predicted probability $p \in (0,1)$. Since they occur with equal probability and have opposite binary targets, their average BCE risk is

$$\mathcal{R}_{\mathrm{dir}}(p) = -\frac{1}{2}\log p - \frac{1}{2}\log(1-p) = -\frac{1}{2}\log\big(p(1-p)\big). \tag{25}$$

Since $p(1-p) \le 1/4$, the risk is at least $\log 2$. This bound is attained by the constant zero logit, which gives $p = 1/2$. Therefore,

$$\inf_{\theta_{\mathrm{dir}}} \mathcal{R}_{\mathrm{dir}} = \log 2. \tag{26}$$

For the heterogeneous formulation, fix the encoder and pair-encoder parameters constructed in the proof of Theorem 1, with $\rho = 0$. Let $\mathbf{h}^+ \neq \mathbf{h}^-$ denote the resulting pair representations for the positive and negative targets, respectively, and define $\mathbf{d} = \mathbf{h}^+ - \mathbf{h}^- \neq \mathbf{0}$. For any $M > 0$, parameterize the shared predictor to realize the affine score $\mathbf{w}_M^\top \mathbf{h} + b_M$, where

$$\mathbf{w}_M = \frac{2M}{\|\mathbf{d}\|_2^2}\mathbf{d}, \qquad b_M = -\frac{1}{2}\mathbf{w}_M^\top(\mathbf{h}^+ + \mathbf{h}^-). \tag{27}$$

Then

$$\mathbf{w}_M^\top \mathbf{h}^+ + b_M = M, \qquad \mathbf{w}_M^\top \mathbf{h}^- + b_M = -M. \tag{28}$$

With $\rho = 0$, both branches use the same pair representation, and their matching weights sum to one. Thus, these are also the final logits after branch combination. Hence the average BCE risk is

$$\mathcal{R}_{\mathrm{het}}(M) = \frac{1}{2}\log(1+\exp(-M)) + \frac{1}{2}\log(1+\exp(-M)) = \log(1+\exp(-M)). \tag{29}$$

As $M \to \infty$, $\mathcal{R}_{\mathrm{het}}(M) \to 0$. Since BCE is nonnegative, it follows that

$$\inf_{\theta_{\mathrm{het}}} \mathcal{R}_{\mathrm{het}} = 0. \tag{30}$$

Combining Eqs. 26 and 30 proves Eq. 24. □

A.1.2 Proof of Theorem 2

We first derive a sufficient condition for the matching margin in Theorem 2, and then prove the BCE-risk comparison. Throughout this section, the candidate label c is fixed, and its subscript is omitted when unambiguous.

With these quantities held fixed, $\mathbf{h}_{ic}$ and $\mathbf{q}_i$ remain random through the sampled node. All probabilities and expectations are taken under this target distribution. Only the group prompts, their projection, the gates, and the shared predictor are optimized under each fusion rule. All logarithms are natural.

Population relation vectors. For the sufficient-condition derivation, assume $\pi = \Pr(Y_{ic} = 1) \in (0,1)$ and define

$$\boldsymbol{\mu}_y = \mathbb{E}[\mathbf{q}_i \mid Y_{ic} = y], \qquad y \in \{0,1\}, \tag{31}$$

and the population deviation

$$\boldsymbol{\delta} = \boldsymbol{\mu}_1 - \mathbb{E}[\mathbf{q}_i] = (1-\pi)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0). \tag{32}$$

Suppose both $\boldsymbol{\delta}_+ = [\boldsymbol{\delta}]_+$ and $\boldsymbol{\delta}_- = [-\boldsymbol{\delta}]_+$ are nonzero. Their normalized affinity and repulsion vectors are

$$\mathbf{a} = \frac{\boldsymbol{\delta}_+}{\|\boldsymbol{\delta}_+\|_1}, \qquad \mathbf{r} = \frac{\boldsymbol{\delta}_-}{\|\boldsymbol{\delta}_-\|_1}. \tag{33}$$

For this sufficient-condition derivation, assume that both training profiles are nonzero and that their normalized versions from Eq. 7 satisfy

$$\max\left\{\|\bar{\mathbf{s}}_c^{\mathrm{aff}} - \mathbf{a}\|_1, \|\bar{\mathbf{s}}_c^{\mathrm{rep}} - \mathbf{r}\|_1\right\} \le \kappa. \tag{34}$$

The error κ includes both finite-sample estimation error and any discrepancy between the edge-weighted training statistics and the target pair distribution.

For $\mathbf{q}_i \neq \mathbf{0}$, its normalization cancels in Eq. 7; for $\mathbf{q}_i = \mathbf{0}$, the fallback weight applies. Hence the same matching weight can be written as

$$\alpha_{ic}^{\mathrm{aff}} = \begin{cases} \dfrac{\langle \mathbf{q}_i, \bar{\mathbf{s}}_c^{\mathrm{aff}} \rangle}{\langle \mathbf{q}_i, \bar{\mathbf{s}}_c^{\mathrm{aff}} + \bar{\mathbf{s}}_c^{\mathrm{rep}} \rangle}, & \langle \mathbf{q}_i, \bar{\mathbf{s}}_c^{\mathrm{aff}} + \bar{\mathbf{s}}_c^{\mathrm{rep}} \rangle > 0, \\ \dfrac{1}{2}, & \text{otherwise.} \end{cases} \tag{35}$$

Separation at the conditional means. For $y \in \{0, 1\}$, define

$$A_y = \langle \boldsymbol{\mu}_y, \mathbf{a} \rangle, \qquad B_y = \langle \boldsymbol{\mu}_y, \mathbf{r} \rangle, \qquad D_y = A_y + B_y. \tag{36}$$

By Eqs. 32 and 33,

$$\begin{aligned} u &:= A_1 - A_0 = \frac{\|\boldsymbol{\delta}_+\|_2^2}{(1-\pi)\|\boldsymbol{\delta}_+\|_1} > 0, \\ w &:= B_0 - B_1 = \frac{\|\boldsymbol{\delta}_-\|_2^2}{(1-\pi)\|\boldsymbol{\delta}_-\|_1} > 0. \end{aligned} \tag{37}$$

All A_y and B_y are nonnegative. Moreover, $A_1 = A_0 + u > 0$ and $B_0 = B_1 + w > 0$, so $D_0, D_1 > 0$.

The gap between the population matching weights is

$$g := \frac{A_1}{D_1} - \frac{A_0}{D_0} = \frac{A_1 B_0 - A_0 B_1}{D_1 D_0} = \frac{u B_0 + A_0 w}{D_1 D_0} > 0. \tag{38}$$

Set

$$\tau_c = \frac{1}{2}\left(\frac{A_1}{D_1} + \frac{A_0}{D_0}\right), \qquad d_{\min} = \min\{D_0, D_1\}, \qquad \eta = \frac{d_{\min} g}{8}. \tag{39}$$

Since the matching weights belong to $[0, 1]$, we have $\tau_c \in [0, 1]$ and $0 < g \leq 1$.

Stability under neighborhood and relation-estimation errors. Suppose $\kappa < \eta$ and consider the event

$$\mathcal{A} = \{\|\mathbf{q}_i - \boldsymbol{\mu}_{Y_{ic}}\|_\infty \leq \eta - \kappa\}. \tag{40}$$

Write

$$\widehat{A} = \langle \mathbf{q}_i, \bar{\mathbf{s}}_c^{\mathrm{aff}} \rangle, \qquad \widehat{B} = \langle \mathbf{q}_i, \bar{\mathbf{s}}_c^{\mathrm{rep}} \rangle. \tag{41}$$

On $\mathcal{A}$, conditional on $Y_{ic} = y$,

$$\begin{aligned} |\widehat{A} - A_y| &\leq \left|\langle \mathbf{q}_i - \boldsymbol{\mu}_y, \bar{\mathbf{s}}_c^{\mathrm{aff}} \rangle\right| + \left|\langle \boldsymbol{\mu}_y, \bar{\mathbf{s}}_c^{\mathrm{aff}} - \mathbf{a} \rangle\right| \\ &\leq \|\mathbf{q}_i - \boldsymbol{\mu}_y\|_\infty \|\bar{\mathbf{s}}_c^{\mathrm{aff}}\|_1 + \|\boldsymbol{\mu}_y\|_\infty \|\bar{\mathbf{s}}_c^{\mathrm{aff}} - \mathbf{a}\|_1 \\ &\leq \eta. \end{aligned} \tag{42}$$

The same argument gives $|\widehat{B} - B_y| \leq \eta$. Consequently,

$$\widehat{A} + \widehat{B} \geq D_y - 2\eta > 0, \tag{43}$$

where the last inequality follows from $\eta = d_{\min} g / 8 \leq D_y / 8$. Thus the fallback case in Eq. 35 does not occur on $\mathcal{A}$.

The matching-weight error satisfies

$$\begin{aligned} \left|\alpha_{ic}^{\mathrm{aff}} - \frac{A_y}{D_y}\right| &= \frac{\left|B_y(\widehat{A} - A_y) - A_y(\widehat{B} - B_y)\right|}{D_y(\widehat{A} + \widehat{B})} \\ &\leq \frac{(A_y + B_y)\eta}{D_y(D_y - 2\eta)} \\ &= \frac{\eta}{D_y - 2\eta} \\ &\leq \frac{2\eta}{d_{\min}} \\ &= \frac{g}{4}. \end{aligned} \tag{44}$$

At the conditional means, the distances to τ_c are both $g/2$. Therefore, on $\mathcal{A}$,

$$(2Y_{ic} - 1)\big(\alpha_{ic}^{\mathrm{aff}} - \tau_c\big) \geq \frac{g}{4}. \tag{45}$$

An explicit margin-error bound. Define the neighborhood variation

$$v = \mathbb{E}\left[\|\mathbf{q}_i - \boldsymbol{\mu}_{Y_{ic}}\|_\infty^2\right]. \tag{46}$$

By Markov's inequality,

$$\Pr(\mathcal{A}^c) \le \min\left\{1, \frac{v}{(\eta-\kappa)^2}\right\}. \tag{47}$$

Hence, whenever $v/(\eta-\kappa)^2 < 1$, the matching-margin condition in Theorem 2 holds with τ_c chosen in Eq. 39 and

$$\gamma_c = \frac{g}{4}, \qquad \varepsilon_c = \frac{v}{(\eta-\kappa)^2}. \tag{48}$$

This establishes the matching-margin condition directly from the relation vectors, neighborhood variation, and estimation error. The argument imposes no sign restriction on the candidate label's own coordinate of $\boldsymbol{\delta}$.

Proof of Theorem 2. We now assume the matching-margin condition in Theorem 2. The following argument applies to any τ_c, γ_c, and ε_c satisfying that condition, independently of the sufficient conditions derived above.

1) Realizing a margin-based fused logit. Because $\mathbf{h}_{ic}$ is bounded, choose $B > 0$ such that $|[\mathbf{h}_{ic}]_1| \le B$ almost surely. Set the gating parameters to $\mathbf{W}_g = \mathbf{0}$ and $\mathbf{b}_g = \mathbf{0}$. Both gating vectors then equal $\frac{1}{2}\mathbf{1}$.

Let $\mathbf{e}_1$ denote the first standard basis vector in the pair-representation space. Choose $M > B$ and prompt parameters satisfying

$$\mathbf{u}_c^{\mathrm{aff}} = \frac{2M}{\rho}\mathbf{e}_1, \qquad \mathbf{u}_c^{\mathrm{rep}} = -\frac{2M}{\rho}\mathbf{e}_1. \tag{49}$$

These are feasible with a trainable prompt projection having a column in the direction $\mathbf{e}_1$ and opposite coefficients in the two selected group prompts. Equation 8 then gives

$$\widetilde{\mathbf{h}}_{ic}^{\mathrm{aff}} = \mathbf{h}_{ic} + M\mathbf{e}_1, \qquad \widetilde{\mathbf{h}}_{ic}^{\mathrm{rep}} = \mathbf{h}_{ic} - M\mathbf{e}_1. \tag{50}$$

Let $a = -M + B$ and $b = M - B$, so $a < b$. For any $\lambda > 0$, choose the shared predictor

$$f_\lambda(\mathbf{z}) = -\lambda\tau_c + \frac{\lambda}{b-a}\left([z_1 - a]_+ - [z_1 - b]_+\right). \tag{51}$$

This function is realized by two ReLU hidden units and an affine output.

For the repulsion branch, $[\widetilde{\mathbf{h}}_{ic}^{\mathrm{rep}}]_1 \le a$, whereas for the affinity branch, $[\widetilde{\mathbf{h}}_{ic}^{\mathrm{aff}}]_1 \ge b$. Thus,

$$f_\lambda(\widetilde{\mathbf{h}}_{ic}^{\mathrm{rep}}) = -\lambda\tau_c, \qquad f_\lambda(\widetilde{\mathbf{h}}_{ic}^{\mathrm{aff}}) = \lambda(1-\tau_c). \tag{52}$$

Using the actual matching weights, the final fused logit is therefore

$$s_\lambda = \alpha_{ic}^{\mathrm{aff}}\lambda(1-\tau_c) + (1-\alpha_{ic}^{\mathrm{aff}})(-\lambda\tau_c) = \lambda(\alpha_{ic}^{\mathrm{aff}} - \tau_c). \tag{53}$$

2) Bounding the matching-based BCE. The BCE for a target Y_{ic} and logit s can be written as

$$\ell(Y_{ic}, s) = \log\left(1 + \exp\big(-(2Y_{ic}-1)s\big)\right). \tag{54}$$

On the matching-margin event in Theorem 2,

$$\ell(Y_{ic}, s_\lambda) \le \log(1 + \exp(-\lambda\gamma_c)). \tag{55}$$

On its complement, $\alpha_{ic}^{\mathrm{aff}}, \tau_c \in [0,1]$ implies $|s_\lambda| \le \lambda$, and hence

$$\ell(Y_{ic}, s_\lambda) \le \log(1 + \exp(\lambda)). \tag{56}$$

The probability of this complement is at most ε_c. Consequently,

$$\begin{aligned}\mathcal{R}^*_{\mathrm{match},c} &\le \mathbb{E}[\ell(Y_{ic}, s_\lambda)] \\ &\le \log(1 + \exp(-\lambda\gamma_c)) + \varepsilon_c \log(1 + \exp(\lambda)) \\ &\le \exp(-\lambda\gamma_c) + \varepsilon_c(\lambda + \log 2).\end{aligned} \tag{57}$$

If $0 < \varepsilon_c < 1$, choose $\lambda = \gamma_c^{-1}\log(1/\varepsilon_c) > 0$. Then

$$\mathcal{R}^*_{\mathrm{match},c} \le \varepsilon_c\left(1 + \log 2 + \frac{1}{\gamma_c}\log\frac{1}{\varepsilon_c}\right) = \Psi(\varepsilon_c, \gamma_c). \tag{58}$$

If $\varepsilon_c = 0$, taking $\lambda \to \infty$ in Eq. 57 gives $\mathcal{R}^*_{\mathrm{match},c} = 0 = \Psi(0, \gamma_c)$.

3) Bounding the uniform-fusion BCE. For the fixed candidate label c, replacing the matching weights by $1/2$ yields a prediction that is a function of $\mathbf{h}_{ic}$ alone. This remains true for every choice of the prompt, gating, projection, and predictor parameters.

Let $\zeta(\mathbf{h}_{ic}) = \Pr(Y_{ic} = 1 \mid \mathbf{h}_{ic})$. For any measurable probability predictor $q(\mathbf{h}_{ic}) \in (0, 1)$, the conditional BCE decomposition gives

$$\begin{aligned}\mathbb{E}[\ell(Y_{ic}, \operatorname{logit} q(\mathbf{h}_{ic}))] &= H(Y_{ic} \mid \mathbf{h}_{ic}) + \mathbb{E}\left[D_{\mathrm{KL}}\left(\operatorname{Ber}(\zeta(\mathbf{h}_{ic}))\| \operatorname{Ber}(q(\mathbf{h}_{ic}))\right)\right] \\ &\geq H(Y_{ic} \mid \mathbf{h}_{ic}).\end{aligned} \tag{59}$$

Here, $\operatorname{Ber}(p)$ denotes the Bernoulli distribution with success probability p, and D_{KL} denotes the Kullback–Leibler divergence. This is the entropy decomposition of logarithmic loss (Gneiting & Raftery, 2007). Taking the infimum over the uniform-fusion parameterizations yields

$$\mathcal{R}^*_{\mathrm{uniform},c} \geq H(Y_{ic} \mid \mathbf{h}_{ic}). \tag{60}$$

Combining Eqs. 58 and 60, including the case $\varepsilon_c = 0$, gives

$$\mathcal{R}^*_{\mathrm{uniform},c} - \mathcal{R}^*_{\mathrm{match},c} \geq H(Y_{ic} \mid \mathbf{h}_{ic}) - \Psi(\varepsilon_c, \gamma_c). \tag{61}$$

The right-hand side is strictly positive whenever $\Psi(\varepsilon_c, \gamma_c) < H(Y_{ic} \mid \mathbf{h}_{ic})$, which proves the theorem. □

A.1.3 Proof of Theorem 3

Proof. All risks and information quantities below are evaluated under the same target pair distribution for the fixed candidate label c. The encoders and training-derived relation statistics are fixed, as in Theorem 2. All logarithms are natural.

1) Lower bound for data-node-only prediction. By definition, every direct predictor under comparison has a predicted probability $q(X_c)$. Let $\eta_X(X_c) = \Pr(Y_{ic} = 1 \mid X_c)$. The entropy decomposition of logarithmic loss (Gneiting & Raftery, 2007) gives

$$\begin{aligned}&\mathbb{E}\left[-Y_{ic} \log q(X_c) - (1 - Y_{ic}) \log(1 - q(X_c))\right] \\ &= H(Y_{ic} \mid X_c) + \mathbb{E}\left[D_{\mathrm{KL}}\left(\operatorname{Ber}(\eta_X(X_c))\| \operatorname{Ber}(q(X_c))\right)\right] \\ &\geq H(Y_{ic} \mid X_c).\end{aligned} \tag{62}$$

Taking the infimum over all admissible direct predictors therefore yields

$$\mathcal{R}^*_{\mathrm{dir},c} \geq H(Y_{ic} \mid X_c). \tag{63}$$

2) Separating the relational information. By the conditional mutual information identity,

$$H(Y_{ic} \mid X_c) = I(Y_{ic}; \mathbf{h}_{ic} \mid X_c) + H(Y_{ic} \mid X_c, \mathbf{h}_{ic}). \tag{64}$$

To interpret the first term, define $\eta_{XH}(X_c, \mathbf{h}_{ic}) = \Pr(Y_{ic} = 1 \mid X_c, \mathbf{h}_{ic})$. Then

$$I_c = \mathbb{E}\left[D_{\mathrm{KL}}\left(\operatorname{Ber}(\eta_{XH}(X_c, \mathbf{h}_{ic}))\| \operatorname{Ber}(\eta_X(X_c))\right)\right]. \tag{65}$$

Thus $I_c > 0$ precisely when the heterogeneous representation changes the conditional target probability on a set of positive probability.

The assumption $I_c > 0$ requires the heterogeneous representation to provide target information beyond X_c. It complements the representational separation in Theorem 1 and is explicitly assumed in the combined theorem. For the message-passing setting of that theorem, X_c may be taken as the attributed rooted L-hop neighborhood up to isomorphism, together with the fixed candidate label.

3) Upper bound for the complete formulation. The prompt and shared-predictor construction in the proof of Theorem 2 establishes

$$\mathcal{R}^*_{\mathrm{match},c} \leq \Psi(\varepsilon_c, \gamma_c). \tag{66}$$

Combining Eqs. 63–64 with Eq. 66 gives

$$\begin{aligned}\mathcal{R}^*_{\mathrm{dir},c} - \mathcal{R}^*_{\mathrm{match},c} &\geq H(Y_{ic} \mid X_c) - \Psi(\varepsilon_c, \gamma_c) \\ &= I_c + H(Y_{ic} \mid X_c, \mathbf{h}_{ic}) - \Psi(\varepsilon_c, \gamma_c) \\ &= I_c + \delta_c > 0.\end{aligned} \tag{67}$$

4) Comparison with uniform fusion. For the fixed candidate label, every uniform-fusion prediction is a function of $\mathbf{h}_{ic}$. The same logarithmic-loss decomposition therefore gives

$$\mathcal{R}^*_{\text{uniform},c} \geq H(Y_{ic} \mid \mathbf{h}_{ic}). \tag{68}$$

Since conditioning cannot increase conditional entropy,

$$H(Y_{ic} \mid \mathbf{h}_{ic}) \geq H(Y_{ic} \mid X_c, \mathbf{h}_{ic}). \tag{69}$$

Using Eq. 66, we obtain

$$\mathcal{R}^*_{\text{uniform},c} - \mathcal{R}^*_{\text{match},c} \geq H(Y_{ic} \mid X_c, \mathbf{h}_{ic}) - \Psi(\varepsilon_c, \gamma_c) = \delta_c > 0. \tag{70}$$

Together, these inequalities prove Eq. 13. □

A.2 Implementation Details

A.2.1 Model Architecture

The whole training paradigm is presented in Algorithm 1.

Algorithm 1 Training procedure of PHeLP.

Require: Data graph $\mathcal{G}_x$, node features $\mathbf{X}$, training-label incidence matrix $\mathbf{Y}^{\text{tr}}$, supervised training pairs Ω_{tr}, label texts $\mathcal{T}_y$, maximum number of epochs E
Ensure: Trained parameters θ^*

Construct the heterogeneous graph as described in Section 3.1, with data–label associations and reverse edges given by Eq. 1;
Initialize label-node features using Eq. 2;
Compute affinity and repulsion relation vectors using Eq. 5, and group labels separately by cosine similarity;
Compute local observed-label compositions using Eq. 6;
Initialize trainable parameters θ;
for epoch $= 1, 2, \ldots, E$ **do**
    Obtain data-node and label-node embeddings using Eqs. 3 and 4;
    Compute pair-specific matching weights for the candidate pairs using Eq. 7;
    Construct shared pair representations and select affinity and repulsion group prompts as described in Section 3.3;
    Obtain the two prompt-refined pair representations using Eq. 8;
    Apply the shared predictor and fuse the branch logits to obtain association probabilities using Eq. 9;
    Compute the training loss over Ω_{tr} using Eq. 10 and update θ;
end for
Select θ^* using validation performance;
return θ^*

A.2.2 Metric Definitions

We report four statistics to characterize the homophily patterns and multi-label characteristics of each dataset. We compute these statistics on the preprocessed data graph $\mathcal{G}_x = (\mathcal{V}_x, \mathcal{E}_{xx})$. Let Y_i denote the ground-truth positive label set of node v_i, and let $\mathcal{N}(i)$ denote its neighborhood.

Edge homophily (Edge-H). Following Zhao et al. (2023), the label similarity between two connected nodes v_i and v_j is measured by the Jaccard similarity:

$$J(i,j) = \frac{|\mathcal{Y}_i \cap \mathcal{Y}_j|}{|\mathcal{Y}_i \cup \mathcal{Y}_j|}. \tag{71}$$

Edge-H is defined as the average label similarity over all edges:

$$H_{\text{edge}} = \frac{1}{|\mathcal{E}_{xx}|} \sum_{(v_i,v_j)\in\mathcal{E}_{xx}} J(i,j). \tag{72}$$

A larger Edge-H indicates stronger homophily between connected nodes.

Node homophily (Node-H). Node-H first averages the label similarity between each node and its neighbors and then averages over all non-isolated nodes:

$$H_{\text{node}} = \frac{1}{|\mathcal{V}_x^+|} \sum_{v_i \in \mathcal{V}_x^+} \frac{1}{|\mathcal{N}(i)|} \sum_{v_j \in \mathcal{N}(i)} J(i,j). \tag{73}$$

where $\mathcal{V}_x^+ = \{v_i \in \mathcal{V}_x : |\mathcal{N}(i)| > 0\}$. Unlike Edge-H, which assigns equal weight to each edge, Node-H assigns equal weight to each non-isolated node.

Label homophily (Label-H). Label-H measures, for each positive label of a node, the proportion of its neighbors that share the same label:

$$H_{\text{label}} = \frac{1}{|\mathcal{V}_x^*|} \sum_{v_i \in \mathcal{V}_x^*} \frac{1}{|Y_i|} \sum_{c \in Y_i} \frac{1}{|\mathcal{N}(i)|} \sum_{v_j \in \mathcal{N}(i)} \mathbb{I}\{c \in Y_j\}. \tag{74}$$

where $\mathcal{V}_x^* = \{v_i \in \mathcal{V}_x : |\mathcal{N}(i)| > 0,\ |Y_i| > 0\}$. A larger Label-H indicates that a node's positive labels are more frequently shared by its neighbors.

Multi-label ratio. The multi-label ratio is defined as the proportion of nodes associated with at least two positive labels:

$$R_{\text{multi}} = \frac{1}{|\mathcal{V}_x|} \sum_{v_i \in \mathcal{V}_x} \mathbb{I}\{|Y_i| \geq 2\}. \tag{75}$$

It characterizes the prevalence of multi-label nodes in a dataset.

A.2.3 Prompts for Text Enhancement with LLM

Table 3 summarizes the prompts used for different datasets. We use dataset-specific contextual information to guide the LLM in generating semantically informative text. Each prompt specifies the dataset domain, node type, label semantics, and target label. The LLM is then instructed to generate a concise description explaining the meaning of the target label, the types of nodes typically associated with it, and the key semantic cues that distinguish it from related labels.

A.2.4 Detailed Hyperparameters

Table 4 summarizes the main dataset-specific hyperparameters used in our experiments. For the baseline methods, their default parameter configurations were used. The embedding dimension of the final node and label representations is fixed to 128 across all datasets.

A.2.5 Label-split protocol

Following Sun et al. (2025), we independently partition the data nodes for each label into training, validation, and test subsets according to the specified ratio. Thus, different labels of the same node may belong to different subsets. Only positive data–label associations assigned to the training split are included in the heterogeneous graph. All annotation-derived quantities, including label co-occurrence relations, affinity and repulsion profiles, and local neighborhood label compositions, are computed exclusively from the training label-incidence matrix. Validation and test labels are used only for model selection and final evaluation, respectively; they do not participate in message passing or neighborhood-relation estimation.

A.2.6 Detailed Datasets

We conduct experiments on six public multi-label graph datasets, covering co-authorship, social, citation, and biological interaction networks. The detailed dataset statistics are reported in Table 5. The datasets are described as follows.

- DBLP (Akujuobi et al., 2019) is a multi-label co-authorship network, where each node represents an author and each edge denotes a co-authorship relation. Each author can be associated with one or more of four research areas: database, data mining, artificial intelligence, and information retrieval. The node attributes are constructed from the titles of papers published by the corresponding author.

Table 3: Prompts used for LLM-based label text enhancement on different datasets.

Dataset	Prompt
HumLoc	You are generating a semantic description for a label. Dataset: humloc Domain: protein-protein interaction network Node type: protein Label meaning: subcellular location Target label: ⟨label⟩ Write one concise paragraph describing this label. Focus on: 1. what this label means; 2. what kinds of protein nodes are usually associated with it; 3. key semantic cues that distinguish it from related labels.
EukLoc	You are generating a semantic description for a label. Dataset: EukLoc Domain: protein-protein interaction network Node type: protein Label meaning: subcellular location Target label: ⟨label⟩ Write one concise paragraph describing this label. Focus on: 1. what this label means; 2. what kinds of protein nodes are usually associated with it; 3. key semantic cues that distinguish it from related labels.
DBLP	You are generating a semantic description for a label. Dataset: dblp Domain: citation / co-authorship network Node type: author Label meaning: research area Target label: ⟨label⟩ Write one concise paragraph describing this label. Focus on: 1. what this label means; 2. what kinds of author nodes are usually associated with it; 3. key semantic cues that distinguish it from related labels.

Table 4: Dataset-specific hyperparameter configurations used for PHeLP.

Hyperparameter	HumLoc	EukLoc	DBLP	BlogCat	PCG	Delve
Hidden dimension	64	32	256	64	64	64
Output dimension	128	128	128	128	128	128
Number of layers	2	2	2	2	2	2
Dropout	0.2	0.2	0.2	0.2	0.2	0.2
Learning rate	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}

- BlogCat (Tang & Liu, 2009) is a social network collected from the BlogCatalog website. Each node represents a blogger, and edges represent friendship relations between bloggers. The labels correspond to the groups joined by individual bloggers, resulting in 39 possible labels. Since the original dataset does not provide intrinsic node attributes, we use a fixed feature initialization in our experiments.
- PCG (Zhao et al., 2023) is a biological interaction network constructed for protein phenotype prediction. Each node represents a protein, and edges correspond to known interactions between proteins. Each protein can be associated with one or more of 15 phenotype

Table 5: Statistics of datasets used in our experiments.

Dataset	#Nodes	#Edges	#Labels	Edge-H	Node-H	Label-H	Multi-label Ratio
DBLP	28,702	68,335	4	0.76	0.81	0.95	0.15
BlogCat	10,312	667,966	39	0.10	0.15	0.19	0.28
PCG	3,233	37,352	15	0.17	0.19	0.31	0.46
HumLoc	3,106	18,496	14	0.40	0.37	0.42	0.18
EukLoc	7,766	13,818	22	0.43	0.41	0.45	0.18
Delve	1,229,280	4,322,275	20	0.52	0.54	0.61	0.19

labels. The dataset provides 32-dimensional sequence-based features for protein representation.

- HumLoc (Zhao et al., 2023) is a human protein interaction network for multi-label subcellular localization prediction. Each node corresponds to a human protein, and edges represent known protein–protein interactions. A protein can be localized to multiple cellular compartments and is therefore associated with one or more of 14 subcellular-location labels. Each protein is represented by a 32-dimensional sequence-based feature vector.
- EukLoc (Zhao et al., 2023) is a protein interaction network for subcellular localization prediction in eukaryotes. Similar to HumLoc, nodes represent proteins and edges encode known protein–protein interactions. Each protein can be associated with one or more of 22 subcellular-location labels, and 32-dimensional sequence-based features are provided as node attributes.
- Delve (Akujuobi et al., 2019) is a large-scale multi-label citation network, where each node represents a scientific paper, and each edge denotes a citation relation between two papers. Each paper can be associated with one or more of 20 predefined topics in machine learning and data mining. The node attributes consist of the paper title and, when available, its abstract, while citation contexts are provided as edge attributes.

A.2.7 Detailed Baselines

We compare PHeLP with twelve representative methods, including existing multi-label learning and multi-label node classification methods, as well as GNNs specifically designed for heterophilic graphs. The detailed baselines are described as follows:

- GCN-PLAIN (Wang et al., 2023) exploits instance- and label-level relations to refine multi-label supervision. We report its GCN-based variant for comparison with graph-based multi-label node classification methods.
- MLGW (Akujuobi et al., 2019) formulates multi-label node classification as collaborative graph walks performed by multiple label-specific agents. The agents jointly capture the relationships between node labels and graph structure as well as correlations among label-specific prediction tasks.
- ML-GCN (Gao et al., 2019) jointly learns node and label representations to model node–label and label–label correlations in a unified embedding space for multi-label node classification.
- LARN (Xiao et al., 2022) leverages label semantics to guide node representation learning and adaptively captures label correlations, enabling label-aware representations for multi-label prediction.
- LANC (Zhou et al., 2021) introduces label-attentive neighborhood convolution to aggregate graph information and employs label attention to capture dependencies between node representations and individual labels.
- VariMul (Song et al., 2021) jointly learns node and label representations under a variational framework and further models higher-order label correlations to improve multi-label node prediction.
- CorGCN (Bei et al., 2025) addresses feature and topology ambiguity induced by multiple labels through correlation-aware graph decomposition and correlation-enhanced graph convolution, explicitly incorporating label correlations into message passing.
- LIP-GCN (Sun et al., 2025) analyzes label influences in graph propagation and feature transformation and constructs a label influence graph to regulate representation learning. We use its GCN-based variant for comparison.
- H2GCN (Zhu et al., 2020) is designed for heterophilic graphs by separating ego and neighborhood representations and incorporating higher-order neighborhood information.
- GPR-GNN (Chien et al., 2021) learns generalized PageRank coefficients to adaptively combine information propagated over different neighborhood orders, allowing flexible propagation under varying homophily levels.

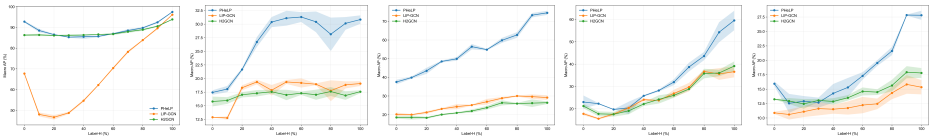
- FAGCN (Bo et al., 2021) adaptively integrates low- and high-frequency graph signals to capture both similar and dissimilar neighborhood information, making it suitable for learning on heterophilic graphs.
- LINKX (Lim et al., 2021) separately encodes graph structure and node features before combining them for prediction, avoiding direct neighborhood feature aggregation.

A.2.8 Efficiency Analysis.

The computational cost of the GraphConv-based heterogeneous encoder is $O\left(L(|\mathcal{E}|d + (N+C)d^2)\right)$, where d is the feature dimension. For each candidate pair, matching requires $O(C)$, while pair encoding, prompt refinement, and prediction require $O(d^2)$ with fixed-depth networks. For M scored candidate pairs, one full-graph encoding followed by pairwise prediction therefore costs $O\left(L(|\mathcal{E}|d + (N+C)d^2) + M(C+d^2)\right)$, excluding preprocessing. During training, $M = |\mathcal{V}_x^{\text{tr}}|C$ under the node-split setting and $M \leq NC$ under the label-split setting.

A.3 Additional Results

A.3.1 Additional Case Study Results

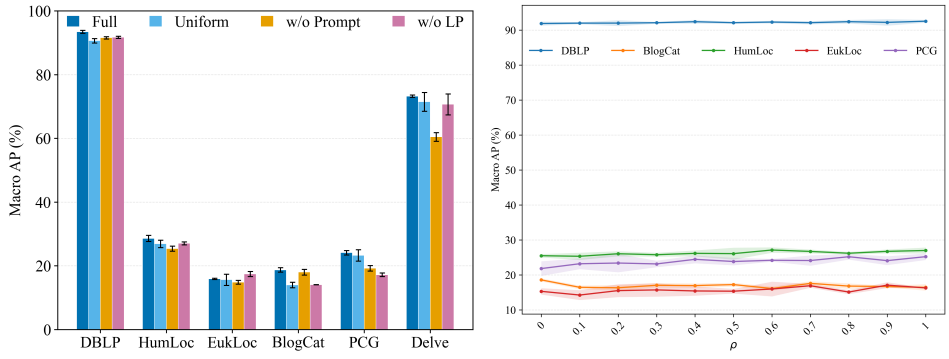


(a) DBLP (b) BlogCat (c) PCG (d) HumLoc (e) EukLoc

Figure 6: Additional case-study results under varying Label-H levels.

Figure 6 shows the Macro AP results under varying Label-H. On BlogCat, PHeLP improves Macro-AP across most levels. On PCG, HumLoc, and EukLoc, its gains are generally larger at higher Label-H. DBLP exhibits a non-monotonic trend: PHeLP remains comparable to H2GCN at intermediate levels while avoiding the sharp deterioration observed for LIP-GCN. These dataset-dependent trends indicate that higher label homophily does not uniformly improve prediction. Overall, PHeLP remains competitive under heterophily while also benefiting from stronger neighborhood label agreement.

A.3.2 Additional Ablation Results on Macro AP.



(a) Ablation results (b) Sensitivity to ρ

Figure 7: Ablation and parameter sensitivity results on Macro AP.

Fig. 7 reports the ablation results in terms of Macro-AP. Full consistently outperforms w/o Prompt across all six datasets, with larger gains on PCG and Delve. Compared with w/o LP, Full achieves higher Macro-AP on five datasets, while w/o LP performs better on EukLoc.

A.3.3 Additional Parameter Sensitivity Results on Macro AP.

Fig. 7b reports the Parameter Sensitivity in terms of Macro AP. Across different values of ρ, the performance remains generally stable, demonstrating that PHeLP is not sensitive to the prompt fusion strength within a broad range.

A.3.4 Effect of Label Semantics

We evaluate the effect of incorporating label semantics through label features and label relations under node split setting with the split of 6:2:2. Emb. denotes augmenting label-node features with PLM embeddings of LLM-enhanced label descriptions, while Rel. augments the label–label co-occurrence graph with each label's top-3 semantic neighbors selected by embedding cosine similarity.

Table 6: Effect of incorporating label semantics into label features and label relations.

Emb.	Rel.	DBLP		HumLoc		EukLoc	
		Macro-AUC	Macro-AP	Macro-AUC	Macro-AP	Macro-AUC	Macro-AP
×	×	96.64 ± 0.25	93.29 ± 0.47	76.67 ± 0.34	28.26 ± 0.90	67.40 ± 12.06	13.00 ± 5.74
×	√	$\mathbf{96.74 \pm 0.17}$	$\mathbf{93.43 \pm 0.44}$	76.84 ± 0.17	28.33 ± 0.20	68.55 ± 12.08	13.14 ± 5.85
√	×	96.60 ± 0.06	93.10 ± 0.08	$\mathbf{77.01 \pm 0.72}$	28.60 ± 0.94	75.94 ± 1.01	$\mathbf{16.56 \pm 0.46}$
√	√	96.57 ± 0.18	92.79 ± 0.35	76.89 ± 0.46	$\mathbf{28.70 \pm 0.75}$	$\mathbf{76.56 \pm 0.25}$	15.90 ± 0.22

Table 6 shows dataset-dependent effects. Semantic relations alone perform best on DBLP, while semantic embeddings yield small gains on HumLoc and substantial improvements on EukLoc. Combining both yields the highest Macro-AP on HumLoc and the highest Macro-AUC on EukLoc, while semantic embeddings alone achieve the highest Macro-AP on EukLoc. Thus, the preferred semantic configuration varies across datasets and metrics.

A.3.5 Prompt Effect across Label-H Groups.

To examine whether the benefit of PHeLP varies with neighborhood label agreement, we conduct a post-hoc analysis on BlogCat. For each label c, we compute $H_c = \frac{1}{|\mathcal{V}^+_{x,c}|} \sum_{v_i \in \mathcal{V}^+_{x,c}} \frac{1}{|\mathcal{N}(i)|} \sum_{v_j \in \mathcal{N}(i)} \mathbb{I}\{c \in \mathcal{Y}_j\}$, where $\mathcal{V}^+_{x,c} = \{v_i \in \mathcal{V}_x : c \in \mathcal{Y}_i, |\mathcal{N}(i)| > 0\}$. We rank the 39 labels by H_c, and divide them into four approximately equal-sized groups from Q1 (lowest Label-H) to Q4 (highest Label-H). For each label, we measure the AUC gain of the prompt-enhanced prediction over the unprompted base prediction from the same trained model, and average the gain within each group.

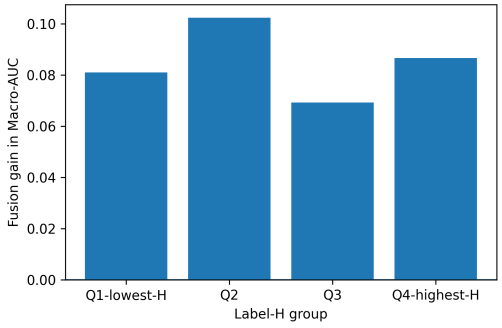


Figure 8: AUC gain of the prompt-enhanced prediction over the unprompted base prediction across Label-H groups on BlogCat.

As shown in Fig. 8, all four groups show positive gains, indicating that PHeLP remains effective across different levels of neighborhood label agreement.