RAISE: RELATIONAL INFERENCE FOR LABEL-EFFICIENT SUBGRAPH CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In biological and social networks, label-efficient subgraph classification predicts labels for designated subgraphs in a shared background graph from a small labeled training subset. In this setting, predictions depend on which labeled subgraphs are informative for a query. Structural similarity alone does not fully capture source–query relations: structurally similar queries can contain different vertices and relate differently to the same labeled source. These relations may help classify queries with no same-class labeled examples among their closest structural matches. We introduce RAISE (Relation-Aligned Inference from Sparse Examples), which encodes aligned membership and diffusion responses on common background-vertex coordinates in a fixed, label-free relational kernel. Kernel ridge regression learns class-specific source coefficients, while class-wise diffusion aggregation supplies complementary evidence. One round of self-training adds high-confidence pseudo-labeled training subgraphs, followed by a dependence-aware refit using the same kernel. Across four benchmarks at 2–20% label budgets, RAISE improves average micro- and macro-F1 over PADEL by 4.71 and 6.74 percentage points, respectively, and achieves the highest aggregate cross-subset agreement. Controlled studies suggest that these relations add predictive information beyond the evaluated structural representations. Code is available here.

1 INTRODUCTION

Many classification problems assign labels to groups of entities in a shared network, such as protein groups in an interaction graph or user communities in a social graph (Alsentzer et al., 2020; Wang & Zhang, 2022). These designated subgraphs may overlap and need not be connected. We study label-efficient subgraph classification, where the background graph and all subgraph memberships are observed, but only a small fraction of training subgraphs are labeled. A central challenge is to make these few labeled examples informative for queries with diverse structural and relational contexts.

Existing methods, including SubGNN (Alsentzer et al., 2020), GLASS (Wang & Zhang, 2022), SSNP (Jacob et al., 2023), and PADEL (Liu et al., 2023), encode subgraph structure, membership, position, and neighborhood context. Other approaches construct inter-subgraph graphs for message passing, as in S2N (Kim & Oh, 2024), or learn cross-subgraph dependencies, as in ISNN (Zhong et al., 2025), while fixed kernels such as WLKS (Kim & Oh, 2025) compare structural patterns within and around designated subgraphs. Structural similarity alone does not fully capture how a query relates to a particular labeled source through shared members and diffusion on the background graph. We focus on the predictive information in these instance-specific relations.

For example, structurally similar subgraphs can contain different vertices and reach different regions of the graph, giving them different membership and diffusion relations to the same labeled source (Figure 1a). Under sparse supervision, a query's class may be represented in the labeled set yet absent among its closest structural matches (a *support-gap query*). In a matched Weisfeiler–Lehman (WL) diagnostic (Xu et al., 2019; Morris et al., 2019) on PPI-BP and HPO-METAB, this gap is linked to 9.98 and 23.73 percentage-point increases in PADEL's prediction error, respectively, holding the query, label budget, and observed class counts fixed (Figure 1b; Appendix G.2). We therefore ask: *do instance-specific relations add predictive evidence beyond structural similarity?*

Our starting point is to record the relations available from the observed graph before fitting the sparse labels. Shared background-vertex coordinates make membership and diffusion responses

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.

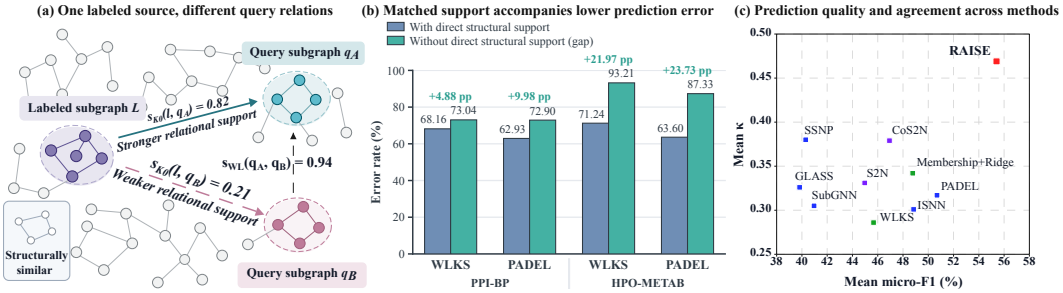


Figure 1: **Relational evidence under sparse supervision.** (a) Structurally similar queries can have different relations to the same labeled source (illustrative scores). (b) Prediction error with and without a support gap, fixing the query, label budget, and observed class counts. (c) Mean test micro-F1 and cross-subset Cohen's κ over four tasks. Panels (b,c) use 2/5/10/20% training labels and five labeled subsets per budget.

directly comparable across subgraphs, and diffusion relates sets even when they share no members. These relations describe how a query is connected to a source, but their strength is not itself a class prediction; the labels must determine how each relation contributes to each class. We therefore separate *relation construction* from *label-conditioned inference*.

We introduce RAISE (Relation-Aligned Inference from Sparse Examples). Its relational kernel is fixed and label-free: aligned membership and diffusion responses are recorded once, and labels act only through inference. Kernel ridge regression (KRR) learns class-specific source coefficients, which enter each query's prediction through its source–query relations, and class-wise diffusion aggregation supplies complementary evidence. One self-training round then adds high-confidence pseudo-labeled examples from the unlabeled training pool and refits with dependence-aware weighting on the same kernel. The observed relations are thus reused as supervision expands, rather than rebuilt for each labeled subset.

Across four benchmarks with 2–20% of training subgraphs labeled, RAISE improves average micro- and macro-F1 over PADEL, the strongest evaluated baseline overall, by 4.71 and 6.74 percentage points, respectively (Table 1). Which subgraphs are labeled usually reflects annotation budget rather than design, so predictions should not hinge on it (Bousquet & Elisseeff, 2002). On cross-subset Cohen's κ, the chance-corrected agreement of predictions across labeled subsets, RAISE achieves the highest aggregate value among the compared methods (Figure 1c). Our contributions are:

- We provide empirical evidence that instance-specific source–query relations contribute predictive information beyond the evaluated structural representations, using matched fitting studies and correspondence interventions.
- We introduce RAISE, which preserves membership and diffusion relations in a fixed, label-free kernel and uses sparse labels for class-specific fitting and one self-training round.
- We demonstrate aggregate F1 and cross-subset agreement gains on four benchmarks at 2–20% label budgets.

2 Related Work

Subgraph representations and inter-subgraph relations. SubGNN separates position, neighborhood, and structural information into dedicated channels (Alsentzer et al., 2020). GLASS uses membership labeling, SSNP pools stochastic subgraph neighborhoods, and PADEL combines position-aware representations with data-efficient augmentation and contrastive learning (Wang & Zhang, 2022; Jacob et al., 2023; Liu et al., 2023). ISNN learns label-aware subgraph representations with implicit dependencies (Zhong et al., 2025). S2N translates designated subgraphs into nodes, and CoS2N adds coarsened virtual subgraphs (Kim & Oh, 2024). These approaches learn per-subgraph embeddings or a relation structure among designated subgraphs. Positional, distance, and labeling encodings add information beyond per-node structure (Srinivasan & Ribeiro, 2020; You et al., 2019; Li et al., 2020; Zhang et al., 2021). RAISE instead keeps membership and diffusion responses on common background-vertex coordinates as a fixed geometry.

Graph kernels and sparse-label inference. Convolution kernels compare structured objects through kernels on their parts (Haussler, 1999), and graph kernels encode connectivity via matrix exponentials or random-walk powers (Kondor & Lafferty, 2002; Smola & Kondor, 2003). WLKS generalizes Weisfeiler–Lehman kernels to designated subgraphs and their neighborhoods, with optional feature kernels (Shervashidze et al., 2011; Kim & Oh, 2025). Kernel ridge regression, manifold regularization, and label propagation fit or propagate sparse labels on a fixed affinity (Saunders et al., 1998; Rasmussen & Williams, 2006; Belkin et al., 2006; Zhu et al., 2003; Zhou et al., 2003), and simple models on fixed propagation can rival graph neural networks (Wu et al., 2019; Huang et al., 2021). RAISE builds its affinity from aligned membership vectors and pooled diffusion responses, then fits class-specific contributions on it. The structural WL kernel is thus a matched control.

Graph self-training under sparse labels. Self-training adds confident predictions on unlabeled examples as supervision (Yarowsky, 1995; Lee, 2013), including on graphs with few labels (Li et al., 2018; Sun et al., 2020). Incorrect pseudo-labels can reinforce errors through confirmation bias (Arazo et al., 2020). Class-balanced selection keeps easy classes from dominating (Zou et al., 2018). DR-GST and DC-GST reweight or select pseudo-labeled nodes to reduce distribution shift, while HC-GST accounts for homophily-ratio distributions in selection and label assignment (Liu et al., 2022; Wang et al., 2024a;b). RAISE uses one train-only round on a fixed relational kernel; its refit models dependence among pseudo sources that share members and pseudo-targets.

3 Problem Formulation

We study label-efficient classification of designated vertex sets in a shared background graph. Let $G = (\mathcal{V}, \mathcal{E})$ be an undirected graph with n vertices, and let $S_1, \dots, S_N \subseteq \mathcal{V}$ be the designated sets to be classified. Each S_i defines an instance through the induced subgraph $G[S_i]$. The sets may overlap and need not be connected. The membership matrix $\mathbf{M} \in \{0,1\}^{N \times n}$ records the vertices in each set. Each instance may have one or more of C class labels, represented by $\mathbf{Y} \in \{0,1\}^{N \times C}$. In single-label tasks, exactly one entry in each row is positive. The graph and all memberships are observed for every instance, while only a subset of training instances is labeled.

Let $\mathcal{D}_{\mathrm{tr}}$, $\mathcal{D}_{\mathrm{val}}$, and $\mathcal{D}_{\mathrm{te}}$ be fixed training, validation, and test index sets. At a given label budget, observed training labels are indexed by $\mathcal{L} \subseteq \mathcal{D}_{\mathrm{tr}}$, and the remaining training instances form the unlabeled pool $\mathcal{U} = \mathcal{D}_{\mathrm{tr}} \setminus \mathcal{L}$. The task is to predict labels for the transductive prediction set $\mathcal{Z} = [N] \setminus \mathcal{L}$, where $[N] = \{1, \dots, N\}$, using the observed graph, memberships, and labels $\mathbf{Y}_{\mathcal{L}}$. We call an instance that supplies a fitting target a *source* and an instance that receives a prediction a *query*. Validation labels select configurations, test labels evaluate predictions, and a label budget specifies the fraction of training labels revealed. Self-training adds pseudo-labeled instances from $\mathcal{U}$ as supervision sources.

4 RAISE: Relational Subgraph Classification

Figure 2 outlines RAISE in three blocks. Block 1 builds membership and diffusion relations on common background-vertex coordinates and combines them into one relational kernel. Block 2 turns sparse labels into class scores by regularized fitting over this kernel and by class-wise aggregation of diffusion responses. Block 3 adds confident predictions from the unlabeled training pool as supervision and performs one dependence-aware refit. The relation matrices remain fixed throughout; supervision changes only how they are used.

4.1 Relational Kernel Construction

Preserving correspondence. A structural description can characterize each subgraph without retaining how its members correspond to those of a particular labeled example. RAISE therefore represents all sets in the same background-vertex coordinates. A bar denotes row-ℓ_2 normalization. For membership, $\langle \bar{\mathbf{M}}_{i,:}, \bar{\mathbf{M}}_{j,:} \rangle = |S_i \cap S_j| / \sqrt{|S_i||S_j|}$, so two sets with identical internal structure can relate differently to a third set. Diffusion responses also relate sets without shared vertices.

Membership and neighborhood responses. Let $\mathbf{A}$ be the raw background adjacency and $\mathbf{P}_{\mathrm{rw}}$ the random-walk (row) normalization of $\mathbf{A} + \mathbf{I}_n$. The rows of $\mathbf{M}\mathbf{P}_{\mathrm{rw}}$ record one random-walk step from

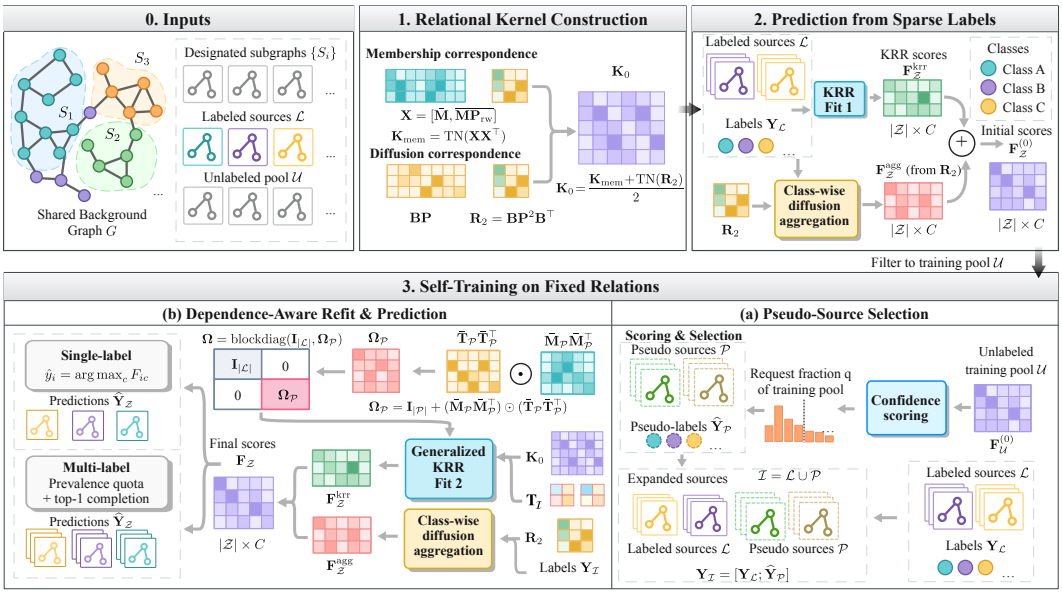


Figure 2: **RAISE overview.** (1) Membership and pooled diffusion responses define the fixed relational kernel $\mathbf{K}_0$. (2) KRR and class-wise diffusion aggregation produce initial class scores from sparse labels. (3a) Confident predictions from the unlabeled training pool supply pseudo sources; (3b) one dependence-aware KRR refit and recomputed diffusion aggregates use the expanded source set. In both fits, the two score branches are separately column-standardized over $\mathcal{Z} = [N] \setminus \mathcal{L}$ and summed, with weight $\lambda = 0.5$ on diffusion aggregation. For final prediction, each selected pseudo source is scored after removing its own supervision from both branches and recomputing normalization moments. Bars denote row-ℓ_2 normalization, $\text{TN}(\mathbf{H}) = N\mathbf{H}/\operatorname{tr}(\mathbf{H})$, and $\mathbf{T}$ denotes encoded fitting targets.

the members. We normalize the membership and neighborhood features separately, concatenate the two views, and form their Gram matrix:

$$\mathbf{X} = [\bar{\mathbf{M}}, \overline{\mathbf{MP}_{\text{rw}}}], \qquad \mathbf{K}_{\text{mem}} = \text{TN}(\mathbf{X}\mathbf{X}^\top), \qquad \text{TN}(\mathbf{H}) = \frac{N\mathbf{H}}{\operatorname{tr}(\mathbf{H})}. \tag{1}$$

Here $\mathbf{X} \in \mathbb{R}^{N \times 2n}$; each block compares responses at matching vertex coordinates, and trace normalization makes the mean diagonal entry one.

Pooled diffusion responses. For weighted members, we use $\mathbf{P} = \mathbf{D}^{-1/2}\widehat{\mathbf{A}}\mathbf{D}^{-1/2}$, where $\widehat{\mathbf{A}}$ is the binarized adjacency with unit diagonal and $\mathbf{D}$ its row sums; symmetry makes the response Gram a two-step relation, whereas $\mathbf{P}_{\text{rw}}$ above supplies random-walk features. The pooling matrix $\mathbf{B} \in \mathbb{R}^{N \times n}$ has unit row sums and $B_{iv} \propto M_{iv} a_v r_{iv}$, with weights defined in Appendix Equation A.1. The weight a_v emphasizes vertices occurring in fewer designated sets, and r_{iv} members with more connectivity beyond the set boundary; both use only memberships and graph structure.

The row $(\mathbf{BP})_{i,:}$ is the pooled one-step diffusion response of S_i. Inner products compare these responses at matching background vertices, giving the two-step relation that we combine with the membership kernel:

$$\mathbf{R}_2 = \mathbf{B}\mathbf{P}^2\mathbf{B}^\top = (\mathbf{BP})(\mathbf{BP})^\top, \qquad \mathbf{K}_0 = \tfrac{1}{2}\big(\mathbf{K}_{\text{mem}} + \text{TN}(\mathbf{R}_2)\big). \tag{2}$$

Both $N \times N$ terms are positive semidefinite Gram matrices with equal weight after trace normalization. Both are label-free, reused across labeled subsets, and unchanged under a joint renumbering of background vertices in the graph and all memberships.

4.2 Prediction from Sparse Labels

Learning class-specific contributions. Labels determine each source's contribution to each class. Let $\mathcal{I}$ index the current m supervision sources, initially $\mathcal{I} = \mathcal{L}$. The targets $\mathbf{T}_\mathcal{I} \in \mathbb{R}^{m \times C}$ are one-hot

labels for single-label tasks and signed labels $2\mathbf{Y}_{\mathcal{I}} - \mathbf{1}_m\mathbf{1}_C^\top$ for multilabel tasks. KRR (Saunders et al., 1998; Rasmussen & Williams, 2006) fits coefficients $\mathbf{W} \in \mathbb{R}^{m\times C}$ and an unpenalized intercept $\mathbf{b} \in \mathbb{R}^C$:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{b}} \ \operatorname{tr}(\mathbf{E}^\top \boldsymbol{\Omega}^{-1}\mathbf{E}) + \alpha \operatorname{tr}(\mathbf{W}^\top \mathbf{K}_{0,\mathcal{I},\mathcal{I}}\mathbf{W}), \\ &\mathbf{E} = \mathbf{T}_{\mathcal{I}} - \mathbf{K}_{0,\mathcal{I},\mathcal{I}}\mathbf{W} - \mathbf{1}_m\mathbf{b}^\top. \end{aligned} \tag{3}$$

We use $\alpha > 0$ and a working residual covariance $\boldsymbol{\Omega} \succ 0$, initially $\mathbf{I}_m$. We solve $(\mathbf{K}_{0,\mathcal{I},\mathcal{I}} + \alpha\boldsymbol{\Omega})\mathbf{W} + \mathbf{1}_m\mathbf{b}^\top = \mathbf{T}_{\mathcal{I}}$ with $\mathbf{1}_m^\top\mathbf{W} = \mathbf{0}^\top$ and compute the fitted scores:

$$\mathbf{F}^{\mathrm{krr}} = \mathbf{K}_{0,:,\mathcal{I}}\mathbf{W} + \mathbf{1}_N\mathbf{b}^\top. \tag{4}$$

The jointly fitted coefficients can have either sign, so the same source–query relation can contribute differently to each class.

Aggregating class-level evidence. To stabilize prediction under sparse supervision, we also average diffusion responses within each class without fitting source-specific coefficients. This branch summarizes $\mathbf{R}_2$, while KRR fits the combined kernel $\mathbf{K}_0$. Let $\mathrm{Std}_{\mathcal{Z}}^{\mathrm{col}}$ standardize each column by its population mean and standard deviation over $\mathcal{Z}$, and put $\widetilde{\mathbf{R}} = \mathrm{Std}_{\mathcal{Z}}^{\mathrm{col}}(\mathbf{R}_{2,:,\mathcal{I}})$. For each class c, let $\mathcal{I}_c^+$ and $\mathcal{I}_c^-$ be the sources with positive and negative supplied labels, let $\chi_{\mathrm{multi}} \in \{0,1\}$ indicate a multilabel task, and let each mean over an empty index set be zero. The class-level aggregates are

$$H_{ic} = \frac{1}{|\mathcal{I}_c^+|}\sum_{j\in\mathcal{I}_c^+}\widetilde{R}_{ij} - \chi_{\mathrm{multi}}\frac{1}{|\mathcal{I}_c^-|}\sum_{j\in\mathcal{I}_c^-}\widetilde{R}_{ij}. \tag{5}$$

We set $\mathbf{F}^{\mathrm{agg}} = \mathrm{Std}^{\mathrm{row}}(\mathbf{H})$ for single-label tasks and $\mathbf{F}^{\mathrm{agg}} = \mathbf{H}$ for multilabel tasks, where $\mathrm{Std}^{\mathrm{row}}$ standardizes each row across classes using population moments.

The complete predictor combines fitted and aggregated scores on a common per-class scale:

$$\mathbf{F} = \mathrm{Std}_{\mathcal{Z}}^{\mathrm{col}}(\mathbf{F}^{\mathrm{krr}}) + \lambda\,\mathrm{Std}_{\mathcal{Z}}^{\mathrm{col}}(\mathbf{F}^{\mathrm{agg}}), \qquad \lambda = 0.5. \tag{6}$$

Both terms participate in initial prediction, pseudo-label selection, and final prediction.

4.3 Self-Training on Fixed Relations

Expanding the supervision sources. The kernel already relates every designated instance, but initially only $\mathcal{L}$ supplies fitting targets. We select confident training predictions and refit on $\mathcal{I} = \mathcal{L}\cup\mathcal{P}$, where $\mathcal{P} \subseteq \mathcal{U}$. Observed labels are retained on $\mathcal{L}$, and predicted binary labels $\widehat{\mathbf{Y}}_{\mathcal{P}}$ supply the new targets, encoded in $\mathbf{T}_{\mathcal{I}}$ by the same rule as above.

Selection requests a fraction q of $\mathcal{U}$. Single-label confidence is the top-two class margin, with class quotas estimated from $\mathcal{L}$. Multilabel confidence measures distance from prevalence-based thresholds, and pseudo-labels are then assigned by class prevalence in the original $\mathcal{L}$. We use one round of self-training. When $\mathcal{P}$ is empty, including at 100% training labels, the initial predictor is final.

Dependence-aware KRR refit. Pseudo sources that share members and a pseudo-label may have correlated errors. We therefore refit Equation 3 with generalized least-squares residual weighting, using the working residual covariance

$$\boldsymbol{\Omega} = \mathrm{blockdiag}\big(\mathbf{I}_{|\mathcal{L}|},\ \mathbf{I}_{|\mathcal{P}|} + (\bar{\mathbf{M}}_{\mathcal{P}}\bar{\mathbf{M}}_{\mathcal{P}}^\top) \odot (\bar{\mathbf{T}}_{\mathcal{P}}\bar{\mathbf{T}}_{\mathcal{P}}^\top)\big)\,. \tag{7}$$

Here $\bar{\mathbf{M}}_{\mathcal{P}}$ and $\bar{\mathbf{T}}_{\mathcal{P}}$ contain row-normalized memberships and encoded pseudo-targets; for single-label tasks, the target Gram is the indicator of equal pseudo-labels. The weighting reduces the penalty along the shared residual directions of such sources. The added term is positive semidefinite by the Schur product theorem, so $\boldsymbol{\Omega}$ is positive definite. Its pseudo block has diagonal two, so disjoint pseudo sources receive half the residual weight of observed sources. The refit changes only the residual weighting in Equation 3; the prediction kernel is unchanged. Class-wise diffusion aggregates are recomputed from the augmented sources and combined as in Equation 6.

Conditional source deletion and prediction. Selected pseudo sources remain in $\mathcal{Z}$ and compete with test queries under multilabel quotas. To avoid scoring a pseudo source by a fit to its own pseudo-label, we remove its supervision from the regression and diffusion aggregation and recompute

its normalization moments; other query rows keep the full-source predictor and its moments. Closed-form KRR leave-one-source-out updates condition on the selected pseudo sources and targets and keep relations and remaining dependence entries fixed (Appendix A.3).

Single-label decisions use argmax. Multilabel decisions use the prevalence-quota rule shared by all compared methods: scores are ranked over the full $\mathcal{Z}$ separately for each class, with positive counts from the original $\mathcal{L}$, and the highest-scoring class fills any remaining empty prediction. Appendix A.4 collects standardization, threshold, quota, and tie conventions. Algorithm A.1 summarizes the procedure.

Computational complexity. Sparse one-step propagation costs $O(N(|\mathcal{E}| + n))$, the response Gram matrices $O(N^2 n)$ in the dense worst case, and their storage $O(N^2)$. With $m = |\mathcal{I}|$ sources, factorization costs $O(m^3)$, fitted scores $O(NmC)$, and the source-deletion influence matrix $O(Nm^2)$. On our $N = 324$–4000 designated sets, relation rebuilding dominates runtime, while cached fits average 0.14 s (Table B.2).

5 Experiments

Following the question in Section 1, we ask whether RAISE improves prediction with sparse labels (Section 5.2), which components carry the gain (Section 5.3), whether the gain rests on instance–relation correspondence rather than structure alone (Section 5.4), and whether predictions stay stable across labeled subsets and missing members (Section 5.5).

5.1 Experimental Setup

Datasets and protocol. We evaluate the biological benchmarks PPI-BP, HPO-METAB, and HPO-NEURO and the social benchmark EM-USER (Alsentzer et al., 2020) (statistics in Table B.1). HPO-NEURO is multilabel; the others are single-label. Training-label budgets are 2/5/10/20%. Methods share the train/validation/test partition, background graph, subgraph memberships, and labeled subsets (Shchur et al., 2018).

Baselines. We compare against the neural subgraph encoders SubGNN (Alsentzer et al., 2020), GLASS (Wang & Zhang, 2022), SSNP (Jacob et al., 2023), PADEL (Liu et al., 2023), and ISNN (Zhong et al., 2025); subgraph-to-node S2N and CoS2N (Kim & Oh, 2024); and fixed-kernel WLKS (Kim & Oh, 2025). Member + 1-hop Ridge is a reference predictor on membership and one-hop features. To separate the kernel from the prediction rule, WL-KRR and $\mathbf{K}_0$-KRR apply the common ridge predictor to the WLKS kernel (internal and background-neighborhood WL views) and to $\mathbf{K}_0$. The suffix + ST adds one self-training round.

Evaluation metrics. Micro-F1 pools class-wise counts before F1, whereas macro-F1 averages class-wise scores. We report test F1 in percent and percentage-point differences, averaging equally over tasks, budgets, and labeled subsets. Cohen's $\kappa = (p_o - p_e)/(1 - p_e)$ (Cohen, 1960) is the chance-corrected agreement. Disagreement is the fraction of unequal classes for single-label tasks and mean label-wise Hamming disagreement for multilabel tasks. Multilabel κ averages binary class-wise values. Both metrics average all run pairs within each task–budget cell; higher κ and lower disagreement indicate greater consistency.

The main results use five labeled-subset draws per budget; the additional analyses in Table 2(b) and Figure 5(a,b) use 50 independently resampled (fresh) subsets per budget. Intervals describe labeled-subset variation with tasks and splits fixed (sampling and interval procedures in Appendices D.2 and E.3).

Implementation details. Validation micro-F1 selects hyperparameters and checkpoints, and baselines retain their representations. All methods share the multilabel prevalence-quota rule, and WLKS keeps its native single-label SVC. RAISE uses one configuration for all tasks and budgets: $\alpha = 0.1$, diffusion weight $\lambda = 0.5$, requested pseudo-label fraction $q = 0.3$, and one self-training round. α is selected once by mean validation micro-F1 over all tasks and six label rates; Table A.2 varies λ and q. In Table 2(a), each KRR variant selects a global α over four low-label rates, whereas panel (b) fixes α and q to RAISE's values. Predictors share column standardization and the task decision rule on $\mathcal{Z}$. Other studies keep the main settings except for stated interventions. Appendix B gives splits and training schedules; Appendix E.1 gives the matched fitting and search settings.

Table 1: Test F1 (%) and cross-subset Cohen's κ, averaged over 2/5/10/20% label budgets using five labeled subsets per budget. Overall columns also average tasks. Bold and underline mark the best and second-best F1 and κ values. Time: total hours over all six label rates (120 conditions per method).

	Micro-F1 by task				Overall			
Method	PPI-BP	METAB	NEURO	EM-USER	Micro	Macro	κ	Time (h)
Member + 1-hop Ridge	43.18	35.40	51.11	65.41	48.77	42.79	0.342	0.03
WLKS	40.31	33.49	48.67	60.20	45.67	38.66	0.286	0.14
SubGNN	37.33	30.13	47.10	49.29	40.96	34.94	0.305	49.81
GLASS	36.23	28.70	45.48	48.88	39.82	34.37	0.326	2.68
SSNP	38.68	20.13	43.30	59.08	40.30	33.57	<u>0.380</u>	1.56
S2N	40.03	38.32	50.29	51.22	44.97	40.06	0.331	2.10
CoS2N	39.28	<u>42.55</u>	50.16	55.71	46.93	42.80	0.379	2.71
ISNN	39.31	37.85	51.58	<u>66.63</u>	48.84	43.01	0.301	2.25
PADEL	<u>43.55</u>	40.89	<u>52.38</u>	65.92	<u>50.69</u>	<u>46.19</u>	0.317	139.60
RAISE	**44.34**	**49.30**	**56.53**	**71.43**	**55.40**	**52.94**	**0.469**	0.22

5.2 Main Results

Prediction quality. RAISE reaches 55.40/52.94 micro/macro-F1, improving on PADEL's 50.69/46.19 by 4.71/6.74 points (Table 1; budget curves and per-task standard deviations in Appendix C). RAISE has the highest budget-averaged micro-F1 on each task, and its task- and budget-averaged gain over PADEL is positive for each of the five subset seeds (95% interval $[4.47, 5.12]$ points).

Agreement across labeled subsets. The F1 advantage comes with greater agreement. RAISE exceeds PADEL and ISNN in all 16 task–budget cells, leads the task-averaged κ at 5, 10, and 20% labels, and has the highest aggregate κ over all budgets (0.469; Appendix D.1).

Computational cost. RAISE builds its label-free relations once per dataset (5–547 s) and fits each labeled condition from the cached matrices in 0.14 s on average. All 120 main-table conditions over six label rates take 0.22 hours, versus 1.56–139.6 hours for the neural baselines (Tables 1 and B.2).

5.3 Component Analysis

Relational kernel. Replacing WL by $\mathbf{K}_0$ helps under all three prediction rules: micro-F1 rises by 4.48 points with class-mean prediction, 5.22 with KRR, and 4.99 with self-training (Table 2a). Diffusion alone, $\mathrm{TN}(\mathbf{R}_2)$ with uniform member pooling, exceeds WL on all four tasks under KRR + ST (mean gain 3.71 points), and weighting by $\mathbf{B}$ adds 0.89 points on average (Appendix E.1). On fresh subsets, $\mathbf{K}_0$ also gains F1 in all eight cells and raises κ from 0.260 to 0.359 (0.267 to 0.397 with self-training; Table 2b).

Support-gap queries. The gains extend to support-gap queries, which have a true class observed in $\mathcal{L}$ yet absent from their five WL-nearest labeled sources (Figure 3). Groups do not use predictions; Appendix F gives the tie-aware definition, coverage, and controls. Over four budgets and five subsets, RAISE exceeds PADEL by 5.65, 14.42, and 5.73 micro-F1 points on PPI-BP, HPO-METAB, and HPO-NEURO, and $\mathbf{K}_0$-KRR + ST (common $\alpha = 0.3$) exceeds WL-KRR + ST by 8.33, 11.91, and 5.98 points on the same tasks; every subset mean is positive in both comparisons.

Class-specific fitting. KRR on $\mathbf{K}_0$ exceeds class-mean prediction by 2.63 points (53.57 versus 50.94; Table 2a), so fitted coefficients add information beyond relation strength.

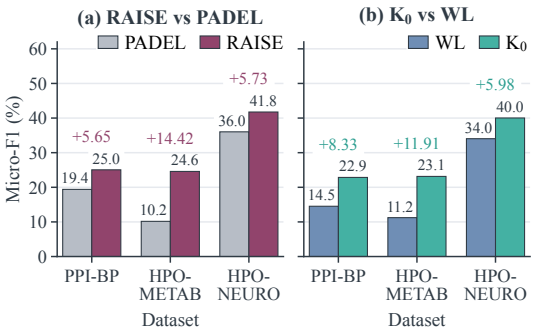


Figure 3: **Prediction on support-gap queries.** Bars show mean micro-F1 over four label budgets and five labeled subsets. Annotations show gains in percentage points.

Table 2: Matched controls averaged over four tasks (F1 in %). (a) Micro-F1 at 2/5/10/20% labels over five original subsets. (b) Results at 5/10% labels over 50 fresh subsets (seeds 1000–1049). Class mean averages each query's similarities to the labeled sources of each class; ST = self-training; agg. = class-wise diffusion aggregation.

(a) Geometry and fitting

Geometry	Class mean	KRR	KRR + ST
WL	46.46	48.35	50.13
Uniform diffusion	45.59	53.24	53.84
Weighted diffusion	50.72	**54.19**	54.73
$\mathbf{K}_0$	**50.94**	53.57	**55.12**

(b) Component sequence

Variant	Micro-F1	Macro-F1	κ
WL-KRR	47.86	45.53	0.260
WL-KRR + ST	49.15	46.64	0.267
$\mathbf{K}_0$-KRR	53.54	51.15	0.359
$\mathbf{K}_0$-KRR + ST	55.39	52.96	0.397
$\mathbf{K}_0$-KRR + ST + agg.	55.74	53.34	**0.462**
RAISE	**55.94**	**53.52**	**0.462**

Self-training. One round improves every geometry in Table 2(a) and, on fresh subsets, adds 1.85 micro-F1 points and 0.038 κ to $\mathbf{K}_0$-KRR.

Aggregation and dependence-aware KRR refitting. Adding class-wise diffusion aggregation to initial prediction, pseudo-source selection, and final prediction raises κ by 0.065 and micro-F1 by 0.35 points, mainly on the biological tasks (Table 2b); the selected pseudo sources change as well. With those sources and targets held fixed, dependence-aware residual weighting adds a further 0.20 points (95% interval $[0.13, 0.27]$; Appendix D.3).

5.4 Instance–Relation Correspondence

We now test whether the gain depends on specific source–query relations rather than on structure alone, by exchanging a labeled source, reassigning relations among instances, and varying relational signal strength (Figure 4; protocols, detailed results, and synthetic recovery studies in Appendices G and H).

Labeled-source exchange. A stronger relation to a single labeled source helps the relational kernel more than WL. On PPI-BP, we swap a labeled source for a WL-matched candidate of the same class that has a stronger diffusion relation to the query. The 733 swaps cover 156 queries, fix the query, label count, and source-class composition, and are selected without query labels or prediction scores. Micro-F1 rises by $+4.89$ points for $\mathbf{K}_0$-KRR and $+0.80$ for WL-KRR (common $\alpha = 0.1$, no self-training), and the $\mathbf{K}_0$-KRR gain is larger in every subset. The query's true-class margin shows the mechanism (Figure 4a): under $\mathbf{K}_0$-KRR, 253 of 254 swaps whose source shares the query's class raise the margin and 375 of 479 other-class swaps lower it, whereas WL-KRR margins stay near zero.

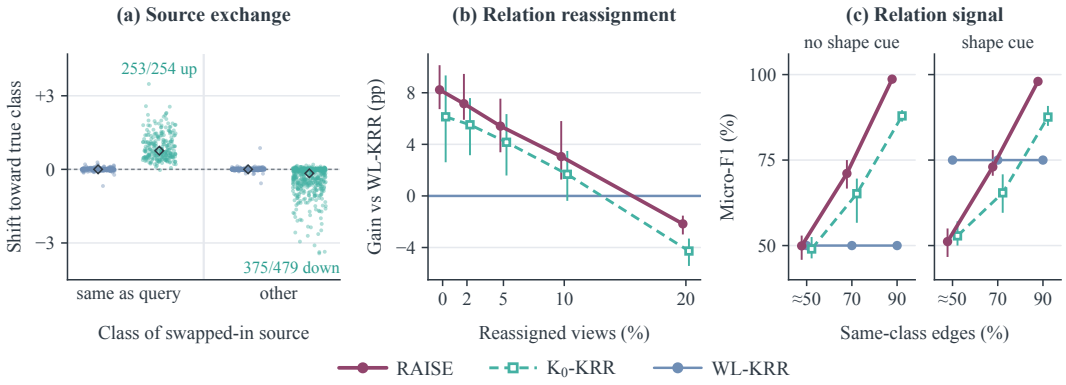


Figure 4: **Effects of source–query correspondence.** (a) Source exchange on PPI-BP (5% labels): change in the query's true-class margin (true-class score minus the best other-class score) for each of 733 swaps, grouped by whether the swapped-in source shares the query's class. Diamonds mark medians. (b) Relation reassignment (four tasks, 10% labels). Zero is WL-KRR. (c) Synthetic tasks with class-independent or informative internal shapes (5% labels). Whiskers show min–max ranges over five labeled subsets (b) or graph seeds (c).

Correspondence reassignment. Prediction degrades as relations are detached from the instances they describe. We permute the kernel rows and columns of 2, 5, 10, and 20% of instances, so that each moved instance keeps its label but takes another instance's relations (labels, splits, parameters, and the full kernel's spectrum, rank, and scale stay fixed; only the pairing changes). RAISE's micro-F1 at 10% labels falls from 59.92 to 58.84, 57.10, 54.74, and 49.52. It stays above WL-KRR up to 10% permuted and falls below it at 20% (Figure 4b).

Relation strength. Synthetic tasks put class information in each designated set's internal shape (path or triangle), in the edges between sets, or both; the same-class fraction of these edges is about 50% (class-independent), 70%, or 90%. At 5% labels, RAISE's micro-F1 rises with this fraction: 49.92, 71.08, and 98.67 with class-independent shapes (WL-KRR: 50.00), and 51.17 and 98.00 at 50% and 90% with informative shapes (WL-KRR: 75.00). RAISE follows the relational signal while WL-KRR stays shape-determined: relations recover labels when structural cues do not (Figure 4c).

5.5 Stability and Robustness

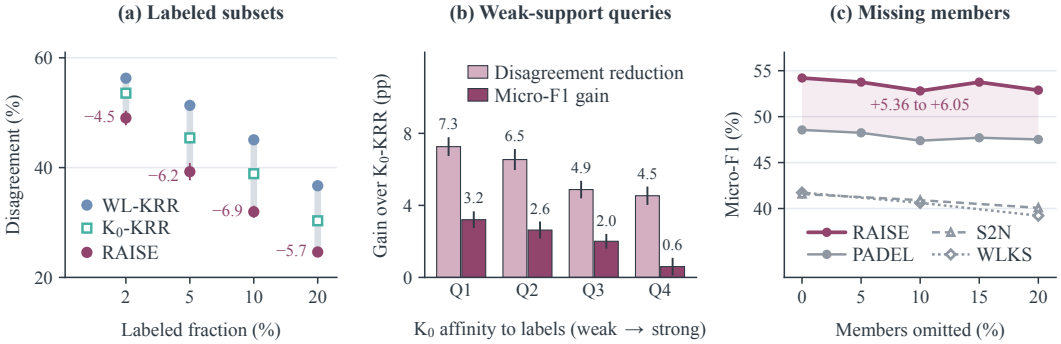


Figure 5: **Stability across labeled subsets and member omission.** All panels average four tasks, and (a,b) use 50 fresh subsets per budget (seeds 1000–1049). (a) Pairwise prediction disagreement. Numbers give RAISE minus $\mathbf{K}_0$-KRR. (b) RAISE's gains over $\mathbf{K}_0$-KRR by maximum $\mathbf{K}_0$ affinity to labeled sources (Q1 weakest). Whiskers show 95% delete-one-subset jackknife intervals. (c) Micro-F1 at 5% labels after rebuilding inputs with 0–20% of members omitted (S2N and WLKS at 0/10/20%). The band marks RAISE's four-task margin over PADEL, positive for every subset seed.

Stability across labeled subsets. RAISE's gains persist on 50 fresh subsets. Its micro-F1 exceeds WL-KRR and $\mathbf{K}_0$-KRR by $+7.08$ and $+2.05$ points (95% intervals $[6.72, 7.45]$ and $[1.84, 2.25]$), positive for every subset seed after averaging over tasks and budgets, and its disagreement is lower than both at every budget (Figure 5a). The disagreement reduction over $\mathbf{K}_0$-KRR holds in every affinity quartile, from 7.26 points for weakest-affinity queries to 4.54 for strongest (Figure 5b).

Robustness to missing members. We remove a random 0/5/10/15/20% of each designated set's members (at least one kept; label-independent masks), and all methods rebuild their inputs from the reduced sets. At 5% labels, RAISE's four-task mean advantage over PADEL stays between 5.36 and 6.05 micro-F1 points as omission lowers both methods' scores (Figure 5c; Appendix I).

6 Conclusion

RAISE classifies subgraphs from membership and diffusion relations on shared vertex coordinates, which preserve member and relation correspondence; sparse labels determine their predictive use, and one self-training round extends supervision. Matched controls and correspondence interventions support the predictive role of these relations. Across four low-label tasks, RAISE has the highest aggregate micro/macro-F1 and cross-subset agreement among the compared methods, supported by fresh-subset and incomplete-observation studies. When class information resides in internal shape rather than in background relations, a structural kernel remains the better choice. Combining the two sources and scaling beyond dense kernels are natural next steps.

Reproducibility statement

The implementation is available in the anonymous code repository linked in the abstract. Appendices A–C specify the method details, evaluation protocol, implementation, and full results. Appendices D–I detail the stability, component, support-gap, correspondence, synthetic, and missing-observation studies with their search settings. Metrics are computed from retained predictions; numerical replay checks reconstruct the frozen predictor from the observed training labels. Each study identifies its labeling rates, subset population, and selection rule.

AI use statement

AI tools assisted with language refinement during manuscript preparation. The authors are responsible for the scientific content and reported results.

References

Emily Alsentzer, Samuel Finlayson, Michelle Li, and Marinka Zitnik. Subgraph neural networks. In *Advances in Neural Information Processing Systems*, pp. 8017–8029, 2020.

Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *Proceedings of the International Joint Conference on Neural Networks*, pp. 1–8, 2020.

Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

Bradley Efron. *The Jackknife, the Bootstrap and Other Resampling Plans*, volume 38 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics, 1982.

David Haussler. Convolution kernels on discrete structures. Technical Report UCSC-CRL-99-10, University of California, Santa Cruz, 1999.

Qian Huang, Horace He, Abhay Singh, Ser-Nam Lim, and Austin R. Benson. Combining label propagation and simple models out-performs graph neural networks. In *International Conference on Learning Representations*, 2021.

Shweta Ann Jacob, Paul Louis, and Amirali Salehi-Abari. Stochastic subgraph neighborhood pooling for subgraph classification. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, pp. 3963–3967, 2023.

Dongkwan Kim and Alice Oh. Translating subgraphs to nodes makes simple GNNs strong and efficient for subgraph representation learning. In *International Conference on Machine Learning*, pp. 24415–24436, 2024.

Dongkwan Kim and Alice Oh. Generalizing Weisfeiler-Lehman kernels to subgraphs. In *International Conference on Learning Representations*, 2025.

Risi Imre Kondor and John Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *International Conference on Machine Learning*, pp. 315–322, 2002.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on Challenges in Representation Learning*, 2013.

Pan Li, Yanbang Wang, Hongwei Wang, and Jure Leskovec. Distance encoding: Design provably more powerful neural networks for graph representation learning. In *Advances in Neural Information Processing Systems*, pp. 4465–4478, 2020.

Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3538–3545, 2018.

Chang Liu, Yuwen Yang, Zhe Xie, Hongtao Lu, and Yue Ding. Position-aware subgraph neural networks with data-efficient learning. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, pp. 643–651, 2023.

Hongrui Liu, Binbin Hu, Xiao Wang, Chuan Shi, Zhiqiang Zhang, and Jun Zhou. Confidence may cheat: Self-training on graph neural networks under distribution shift. In *Proceedings of the ACM Web Conference*, pp. 1248–1258, 2022.

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 *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 4602–4609, 2019.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

Craig Saunders, Alexander Gammerman, and Vladimir Vovk. Ridge regression learning algorithm in dual variables. In *International Conference on Machine Learning*, pp. 515–521, 1998.

Bernhard Schölkopf, Ralf Herbrich, and Alex J. Smola. A generalized representer theorem. In *Proceedings of the Annual Conference on Computational Learning Theory*, pp. 416–426, 2001.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

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.

Alexander J. Smola and Risi Kondor. Kernels and regularization on graphs. In *Proceedings of the Annual Conference on Learning Theory*, pp. 144–158, 2003.

Balasubramaniam Srinivasan and Bruno Ribeiro. On the equivalence between positional node embeddings and structural graph representations. In *International Conference on Learning Representations*, 2020.

Ke Sun, Zhouchen Lin, and Zhanxing Zhu. Multi-stage self-supervised learning for graph convolutional networks on graphs with few labeled nodes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 5892–5899, 2020.

Fali Wang, Tianxiang Zhao, and Suhang Wang. Distribution consistency based self-training for graph neural networks with sparse labels. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, pp. 712–720, 2024a.

Fali Wang, Tianxiang Zhao, Junjie Xu, and Suhang Wang. HC-GST: Heterophily-aware distribution consistency based graph self-training. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, pp. 2326–2335, 2024b.

Xiyuan Wang and Muhan Zhang. GLASS: GNN with labeling tricks for subgraph representation learning. In *International Conference on Learning Representations*, 2022.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International Conference on Machine Learning*, pp. 6861–6871, 2019.

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

David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 189–196, 1995.

Jiaxuan You, Rex Ying, and Jure Leskovec. Position-aware graph neural networks. In *International Conference on Machine Learning*, pp. 7134–7143, 2019.

Muhan Zhang, Pan Li, Yinglong Xia, Kai Wang, and Long Jin. Labeling trick: A theory of using graph neural networks for multi-node representation learning. In *Advances in Neural Information Processing Systems*, pp. 9061–9073, 2021.

Yongjian Zhong, Liao Zhu, Hieu Vu, and Bijaya Adhikari. Implicit subgraph neural network. In *International Conference on Machine Learning*, pp. 78602–78615, 2025.

Dengyong Zhou, Olivier Bousquet, Thomas N. Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems*, pp. 321–328, 2003.

Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In *International Conference on Machine Learning*, 2003.

Yang Zou, Zhiding Yu, B. V. K. Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European Conference on Computer Vision*, pp. 297–313, 2018.

A GEOMETRY AND PREDICTION DETAILS

This appendix specifies the relation construction, fitting, and prediction steps of Section 4; Algorithm A.1 summarizes the complete procedure.

Algorithm A.1 RAISE: prediction and self-training on fixed shared relations.

Require: G, $\mathbf{M}$, observed indices $\mathcal{L}$ and labels $\mathbf{Y}_{\mathcal{L}}$, unlabeled training pool $\mathcal{U}$; $\alpha = 0.1$, $\lambda = 0.5$, $q = 0.3$.
Ensure: Predictions on $\mathcal{Z} = [N] \setminus \mathcal{L}$.
1: Construct $\mathbf{K}_0$ and $\mathbf{R}_2$.
2: Set $\mathcal{I} \leftarrow \mathcal{L}$ and $\mathbf{\Omega} \leftarrow \mathbf{I}_m$; fit on $\mathcal{I}$ and compute the scores in Eq. 6.
3: Select $\mathcal{P} \subseteq \mathcal{U}$ and assign pseudo-labels $\widehat{\mathbf{Y}}_{\mathcal{P}}$.
4: **if** $\mathcal{P} \neq \varnothing$ **then**
5: Form $\mathbf{\Omega}$ and refit on $\mathcal{L} \cup \mathcal{P}$.
6: Recompute both score terms, using conditional deletion for each predicted pseudo-source.
7: **end if**
8: Apply the task decision rule on $\mathcal{Z}$.

A.1 NOTATION AND SOURCE INDEXING

Bold capitals, bold lowercase, and calligraphic letters denote matrices, vectors, and sets, and $\mathbf{1}_r$ and $\mathbf{I}_r$ are the all-ones vector and $r \times r$ identity. We use i, j for designated-instance indices, u, v for background vertices, and c for classes. A vertex set $S_i \subseteq \mathcal{V}$ defines one instance, whereas $\mathcal{L}, \mathcal{U}, \mathcal{P}, \mathcal{I}$, and $\mathcal{Z}$ contain instance indices. For a matrix $\mathbf{H}$, $\mathbf{H}_{\mathcal{A},\mathcal{B}}$ selects rows and columns in the specified index order. When a matrix is restricted to sources, its rows or columns retain the source identifiers. Augmented sources are ordered with $\mathcal{L}$ before $\mathcal{P}$. Unbold subscripts such as M_{iv} and F_{ic} denote scalar entries. Table A.1 summarizes the main objects.

Table A.1: Core notation. Source count $m = |\mathcal{I}|$ changes after self-training, but N and n do not.

Symbol	Type or dimension	Meaning
S_i	subset of $\mathcal{V}$	Vertices of designated subgraph i
$\mathcal{L}, \mathcal{U}, \mathcal{P}$	subsets of $[N]$	Observed, unlabeled-training, and pseudo sources
$\mathcal{I}$	subset of $[N]$	Current sources: $\mathcal{L}$ or $\mathcal{L} \cup \mathcal{P}$
$\mathcal{Z}$	$[N] \setminus \mathcal{L}$	Prediction and column-standardization indices
$\mathbf{M}, \mathbf{B}$	$N \times n$	Membership and row-sum-normalized pooling
$\mathbf{P}_{\mathrm{rw}}, \mathbf{P}$	$n \times n$	Row-normalized and symmetric propagation
$\mathbf{K}_{\mathrm{mem}}, \mathbf{R}_2, \mathbf{K}_0$	$N \times N$	Membership, diffusion, and combined kernels
$\mathbf{Y}_{\mathcal{L}}, \widehat{\mathbf{Y}}_{\mathcal{P}}$	binary label rows	Observed labels and selected pseudo-labels
$\mathbf{T}_{\mathcal{I}}$	$m \times C$	Encoded fitting targets
$\mathbf{W}, \mathbf{b}$	$m \times C$, $C \times 1$	Regression coefficients and intercept
$\mathbf{\Omega}$	$m \times m$	Working residual covariance
$\mathbf{F}^{\mathrm{krr}}, \mathbf{F}^{\mathrm{agg}}, \mathbf{F}$	$N \times C$	Regression, aggregation, and combined scores

The kernel matrices do not depend on the observed labels. The choice of $\mathcal{L}$ determines $\mathcal{Z}$, which stays fixed within a run. Predictions are evaluated on $\mathcal{D}_{\mathrm{te}}$, and only the remaining training indices $\mathcal{U}$ are eligible for pseudo-label selection. We use C_{svm} for SVC regularization, reserving C for the number of classes.

A.2 MEMBERSHIP AND DIFFUSION FEATURES

For membership features, $\mathbf{P}_{\mathrm{rw}}$ row-normalizes the raw adjacency $\mathbf{A}$ after adding $\mathbf{I}_n$. For diffusion responses, the raw adjacency is binarized and its diagonal is set to one, and the resulting $\widehat{\mathbf{A}}$ is symmetrically normalized as $\mathbf{P} = \mathbf{D}^{-1/2}\widehat{\mathbf{A}}\mathbf{D}^{-1/2}$. The membership block uses a random-walk feature map, whereas the symmetry of the diffusion block makes its response Gram equal to $\mathbf{B}\mathbf{P}^2\mathbf{B}^\top$. Because the benchmark graphs are unweighted, the two conventions differ only in normalization and at self-looped vertices, which occur only in PPI-BP. Let $f_v = \sum_i M_{iv}$ count memberships, and let $d_G(v)$ and $d_{S_i}(v)$ be degrees in the raw background and induced-subgraph adjacency. Row-sum-

normalized pooling is defined as follows:

$$B_{iv} = \frac{M_{iv} a_v r_{iv}}{\sum_u M_{iu} a_u r_{iu}}, \qquad a_v = 1 + \log \frac{N+1}{f_v + 1}, \qquad r_{iv} = 1 + \log \frac{d_G(v)+1}{d_{S_i}(v)+1}. \tag{A.1}$$

The sets are nonempty, so these row sums are positive. This pooling normalization differs from the row-ℓ_2 normalization of the two blocks of $\mathbf{X}$. Since $\mathbf{P}$ is symmetric, $\mathbf{R}_2$ has the Gram form $(\mathbf{BP})(\mathbf{BP})^\top$ and is positive semidefinite.

We construct the membership features and pooled diffusion responses with sparse matrix operations. Membership rows and their one-step features are separately normalized before concatenation. The sparse membership features use float32 and are converted to float64 before forming the dense membership Gram. Diffusion responses, trace normalization, and prediction solves use float64. The dense relation matrices are cached across label budgets. Table B.2 separates rebuilding from fitting on these cached matrices.

A joint vertex permutation preserves the relational kernels. For a permutation matrix $\mathbf{\Pi}_\mathrm{v} \in \{0,1\}^{n \times n}$, let $\mathbf{M}' = \mathbf{M}\mathbf{\Pi}_\mathrm{v}^\top$, $\mathbf{B}' = \mathbf{B}\mathbf{\Pi}_\mathrm{v}^\top$, and $\mathbf{P}' = \mathbf{\Pi}_\mathrm{v}\mathbf{P}\mathbf{\Pi}_\mathrm{v}^\top$. Then $\mathbf{B}'(\mathbf{P}')^2\mathbf{B}'^\top = \mathbf{B}\mathbf{P}^2\mathbf{B}^\top$, and the membership Gram is also unchanged. Joint vertex renumbering preserves the correspondence between each instance and its relations.

An eight-vertex cycle gives a simple exact example. With vertices numbered consecutively, $S_s = \{0,1\}$, $S_a = \{1,2\}$, and $S_b = \{5,6\}$ each induce a single edge and have identical WL histograms under uniform initialization. Their normalized membership inner products with S_s are $1/2$ for S_a and 0 for S_b. The membership vectors retain which vertex is shared, so the supervised fit can use these distinct connections.

A.3 Dependence-Aware KRR Refit and Source Deletion

Let $m = |\mathcal{I}|$, $\mathbf{Q} = \mathbf{K}_{0,\mathcal{I},\mathcal{I}} + \alpha\mathbf{\Omega}$, $\mathbf{J} = \mathbf{Q}^{-1}$, $\mathbf{h} = \mathbf{J}\mathbf{1}_m$, and $\eta = \mathbf{1}_m^\top\mathbf{h}$. With the column-vector intercept convention, the solution has the following closed form:

$$\mathbf{b} = \frac{\mathbf{T}_\mathcal{I}^\top\mathbf{h}}{\eta}, \qquad \mathbf{W} = \mathbf{J}\mathbf{T}_\mathcal{I} - \mathbf{h}\mathbf{b}^\top, \qquad \mathbf{F}^{\mathrm{krr}} = \mathbf{K}_{0,:,\mathcal{I}}\mathbf{W} + \mathbf{1}_N\mathbf{b}^\top. \tag{A.2}$$

Here $\mathbf{b} \in \mathbb{R}^C$, $\mathbf{W} \in \mathbb{R}^{m \times C}$, and $\mathbf{1}_m^\top\mathbf{W} = \mathbf{0}^\top$. Equation 3 is the coefficient form of KRR with kernel $k_0(i,j) = (\mathbf{K}_0)_{ij}$. For functions $f_c \in \mathcal{H}_{k_0}$, the representer theorem (Schölkopf et al., 2001) gives $f_c(i) = \sum_{j \in \mathcal{I}} k_0(i,j) W_{jc}$ and $\sum_c \|f_c\|^2_{\mathcal{H}_{k_0}} = \mathrm{tr}(\mathbf{W}^\top\mathbf{K}_{0,\mathcal{I},\mathcal{I}}\mathbf{W})$. The objective weights source residuals by $\mathbf{\Omega}^{-1}$ and leaves the intercept unpenalized.

The initial fit uses Equation A.2 with $\mathbf{\Omega} = \mathbf{I}_m$. In the refit, the added term of the pseudo block is the Hadamard product of the membership and target Gram matrices and is therefore positive semidefinite by the Schur product theorem. Adding the identity makes $\mathbf{\Omega}$ positive definite, and $\alpha > 0$ then makes $\mathbf{Q}$ positive definite even if $\mathbf{K}_0$ is rank deficient; all solves use Cholesky factorizations. The working residual covariance is built from memberships and supplied pseudo-targets, without estimating the true error covariance. Since its pseudo diagonal is two, disjoint pseudo sources receive half the residual penalty of observed sources.

For multilabel pseudo-targets, $\bar{\mathbf{T}}_{j,:} = (2\widehat{\mathbf{Y}}_{j,:} - \mathbf{1}_C^\top)/\sqrt{C}$. The inner product between rows i and j is $1 - 2d_H(\widehat{\mathbf{Y}}_{i,:}, \widehat{\mathbf{Y}}_{j,:})/C$, where d_H is Hamming distance. This similarity counts agreement on negative labels and may be negative, and an all-zero binary pseudo-label row still has a nonzero signed encoding. Single-label rows use one-hot targets, so their inner product indicates equal pseudo-labels.

Define $\mathbf{J}_\perp = \mathbf{J} - \mathbf{h}\mathbf{h}^\top/\eta$ and $\mathbf{H}_* = \mathbf{K}_{0,:,\mathcal{I}}\mathbf{J}_\perp + \mathbf{1}_N\mathbf{h}^\top/\eta$. For source j, set $\boldsymbol{\delta}_j^\top = \mathbf{W}_{j,:}/(\mathbf{J}_\perp)_{jj}$. With at least two sources, deleting that source under the fixed kernel and corresponding principal dependence block gives the updated scores:

$$\mathbf{F}^{\mathrm{krr},-j}_{i,:} = \mathbf{F}^{\mathrm{krr}}_{i,:} - (\mathbf{H}_*)_{ij}\boldsymbol{\delta}_j^\top. \tag{A.3}$$

This closed-form leave-one-source-out update includes the change in fitted coefficients and intercept after source deletion.

For a source j that remains in $\mathcal{Z}$, its score and its column-normalization moments use this conditional deletion: the moments are those of $\mathbf{F}^{\mathrm{krr},-j}_{\mathcal{Z},:}$, computed with the same influence vector as Equation A.3.

Diffusion aggregation likewise removes j from the class sums and counts and recomputes its row and column normalization. The update conditions on a single pseudo-label selection and keeps its instances and targets, the relation matrices, and the remaining dependence entries fixed. Only the source's own score row is replaced; other query rows retain the full-source predictor and its normalization moments. The final decision rule acts on the assembled rows over $\mathcal{Z} = [N] \setminus \mathcal{L}$, including selected pseudo sources. Conditional deletion can therefore change test predictions through multilabel quota ranking, whereas single-label test argmax decisions are unchanged by it.

A.4 Diffusion Aggregation, Confidence, and Prediction Rules

For any score or response matrix $\mathbf{H}$ with instance rows and a generic column index ℓ, define moments on $\mathcal{Z}$ by $\mu_\ell = |\mathcal{Z}|^{-1} \sum_{i \in \mathcal{Z}} H_{i\ell}$ and $\sigma_\ell^2 = |\mathcal{Z}|^{-1} \sum_{i \in \mathcal{Z}} (H_{i\ell} - \mu_\ell)^2$. Then $(\mathrm{Std}_{\mathcal{Z}}^{\mathrm{col}}(\mathbf{H}))_{i\ell} = (H_{i\ell} - \mu_\ell)/\sigma_\ell$ when $\sigma_\ell > 10^{-12}$, and zero otherwise. The operator $\mathrm{Std}^{\mathrm{row}}$ uses the same population-moment convention across classes within each row.

Class aggregation first standardizes each source column of $\mathbf{R}_{2,:,\mathcal{I}}$ over $\mathcal{Z}$; the resulting matrix $\widetilde{\mathbf{R}}$ holds the standardized source–query responses. For each class, we average over positive sources; multilabel tasks subtract the negative-source average, single-label tasks standardize the class averages within each row, and empty means contribute zero. We then column-standardize the class-wise scores over $\mathcal{Z}$, with the same operator that standardizes the KRR scores. Conditional source deletion precedes the final normalization, with moments updated as described above.

For a nonempty pool $\mathcal{U}$, the requested pseudo-label count is $n_{\mathrm{req}} = \max(1, \mathrm{round}(q|\mathcal{U}|))$ with $q = 0.3$. Single-label quotas use Laplace-smoothed frequencies $(n_c + 1)/(|\mathcal{L}| + C)$, where $n_c = \sum_{i \in \mathcal{L}} Y_{ic}$, integer floors, and largest-remainder allocation. Candidates within a predicted class are ordered by their top-two score margin. Unavailable class quota is left unfilled. For multilabel tasks, let $p_c = |\mathcal{L}|^{-1} \sum_{i \in \mathcal{L}} Y_{ic}$. Confidence is the mean absolute distance from thresholds determined by $\mathrm{round}(|\mathcal{U}|p_c)$ positives. After choosing the top n_{req} instances, pseudo-label quotas are recomputed within the selected set using the same prevalence. This intermediate assignment follows the class quotas and permits empty label rows. Ties use ascending instance identity.

Final multilabel prediction ranks all instances in $\mathcal{Z}$ independently for each class, assigning positives to the first $\mathrm{round}(|\mathcal{Z}|p_c)$ instances. Any remaining empty prediction receives its highest-scoring class, so this completion can increase class counts beyond the initial quotas. All prevalence estimates use the original $\mathcal{L}$. Rounding uses nearest-even semantics. Single-label predictions use argmax.

The implementation fixes $\lambda = 0.5$, $q = 0.3$, and one round of self-training, and selects a single α by mean validation micro-F1 over four tasks, six label rates, and five labeled subsets from the candidate set $\{0.003, 0.01, 0.03, 0.1, 0.3, 1, 3\}$; the selected value is 0.1. The RAISE implementation uses Python 3.11.15 and PyTorch 2.5.1+cu121 and was evaluated on an RTX 3090.

Table A.2 varies the two fixed values λ and q for the complete predictor at $\alpha = 0.1$. Across $\lambda \in \{0, 0.25, 0.5, 0.75, 1\}$ and $q \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, mean test micro-F1 over 2/5/10/20% labels ranges from 53.81 to 55.63, compared with 55.40 at the fixed values and 50.69 for PADEL. Selecting (λ, q) by mean validation micro-F1 over all 120 conditions gives $\lambda = 0.5$ and $q = 0.5$, with test micro-F1 55.61.

Table A.2: Sensitivity to the diffusion weight λ and the requested pseudo-label fraction q at $\alpha = 0.1$. Entries are mean Micro-F1 (%) over four tasks, 2/5/10/20% labels, and the original five labeled subsets. The fixed configuration ($\lambda = 0.5$, $q = 0.3$) is underlined.

	Validation Micro-F1					Test Micro-F1				
$\lambda \setminus q$	0.1	0.2	0.3	0.4	0.5	0.1	0.2	0.3	0.4	0.5
0	51.06	51.69	51.84	52.27	52.41	53.81	54.72	54.96	55.16	54.84
0.25	52.16	52.39	53.08	53.24	53.14	54.49	55.23	55.46	55.55	55.63
0.5	52.23	52.50	<u>53.20</u>	53.11	53.39	54.35	55.06	<u>55.40</u>	55.30	55.61
0.75	52.13	52.45	53.07	52.97	53.07	54.28	54.89	55.10	55.15	55.34
1	51.90	52.34	52.77	52.84	52.70	54.17	54.71	54.79	54.79	55.21

B Evaluation Protocol and Implementation

Real-task studies use the five original labeled subsets (seeds 0–4) unless they state that they use the 50 fresh labeled subsets (seeds 1000–1049). Table B.1 lists dataset statistics and the effective splits. Instance identities follow the subgraph files. Following the benchmark loader, the larger raw held-out set serves as validation and the smaller as test (Table B.1), and every method uses the same effective partition. The original subsets are sampled from the training indices, and all methods receive the same subsets; unlabeled training instances remain part of the graph geometry. HPO-METAB and HPO-NEURO use the same background graph with different subgraph tasks.

Table B.1: Dataset statistics and effective splits. The background graph has n vertices and $|\mathcal{E}|$ undirected edges, excluding self-loops (3,691 in PPI-BP). HPO-METAB and HPO-NEURO share it. N designated subgraphs have mean size $\overline{|S_i|}$ and C outputs.

Dataset	n	$\lvert\mathcal{E}\rvert$	N	$\overline{\lvert S_i\rvert}$	Train	Validation	Test	C	Target
PPI-BP	21,521	338,625	1,591	10.2	1,272	160	159	6	Single-label
HPO-METAB	14,587	3,238,174	2,400	14.4	1,921	244	235	6	Single-label
HPO-NEURO	14,587	3,238,174	4,000	14.8	3,204	400	396	10	Multilabel
EM-USER	57,333	4,573,417	324	155.4	226	49	49	2	Single-label

All reported F1 scores are computed from saved predictions. For each method, the evaluation uses labeled-subset seeds 0–4 and method seed 0. Standard deviations are computed across labeled subsets with $\mathrm{ddof} = 0$. At 100% labels the observed subsets coincide, and RAISE reuses the equivalent deterministic prediction. Macro-F1 uses the canonical class space and assigns zero to undefined per-class F1. All methods use the same multilabel prevalence-quota prediction rule, while WLKS preserves its native single-label SVC decision.

We use the training schedules from the baseline implementations. GLASS uses up to 300 epochs with native early stopping. SSNP uses 100 epochs, 20 training views, and five views per loader. ISNN uses 1,500 epochs and dataset-specific fixed-point iterations. PADEL uses 500 PA-VGAE epochs, 100 contrastive epochs, and 100 classifier epochs. S2N and CoS2N use 400 epochs for PPI-BP, 2,500 for the HPO tasks, and 1,200 for EM-USER. SubGNN uses 200 epochs with its dataset-specific channels. Member + 1-hop Ridge and WLKS use their kernel solvers.

Table B.2 reports the cost of the main-table runs over four tasks, six label rates, and five labeled subsets. RAISE constructs its relation matrices once per dataset and then fits each labeled condition from the cached matrices in 0.14 s on average. We take the construction cost as the median end-to-end rebuild time in a matched setting: 5.0, 207.2, 546.8, and 23.6 s on PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER. In that setting, every method rebuilds all membership-dependent inputs from the same observed sets of the 10/20% member-omission study (Appendix I; 40 conditions per method on one RTX 3090 host), and the median per-condition times are 110 s for RAISE, 87 s for WLKS, 155 s for S2N, and 3,061 s for PADEL.

Table B.2: Computational cost of the main-table runs (120 conditions per method). Times are worker wall-clock on shared hosts. Per-condition values are medians, except for RAISE, which reports the mean fit on cached relation matrices. Baseline totals exclude one-time, label-independent preprocessing, whereas the RAISE total includes relation construction. GPU memory is the median peak allocation (RAISE: peak during cached fits).

Method	Device	Per condition (s)	Total (h)	GPU memory (MiB)
Member + 1-hop Ridge	CPU	1	0.03	–
WLKS	CPU	4	0.14	–
SubGNN	RTX 3090	1,254.5	49.81	1,829
GLASS	RTX 3090	55.0	2.68	665
SSNP	RTX 3090	36.5	1.56	707
S2N	RTX 3090	61.5	2.10	284
CoS2N	RTX 3090	71.0	2.71	409
ISNN	RTX 3090	70.0	2.25	925
PADEL	RTX 3090	3,542.5	139.60	476
RAISE	RTX 3090	0.14	0.22	927

C Complete Real-World Results

Tables C.1–C.4 report the complete micro-F1 and macro-F1 results at all six label rates. Each entry is a five-subset mean and population standard deviation, and boldface marks the highest mean within a metric and rate among RAISE and the nine comparison configurations. The task columns of Table 1 average the four main-text rates within each task, and its overall columns also average the tasks equally. Over 5/10/20%, RAISE reaches 58.81/56.36 and PADEL 54.18/50.10. The six-rate averages are 59.36/56.92 and 55.43/51.06, respectively.

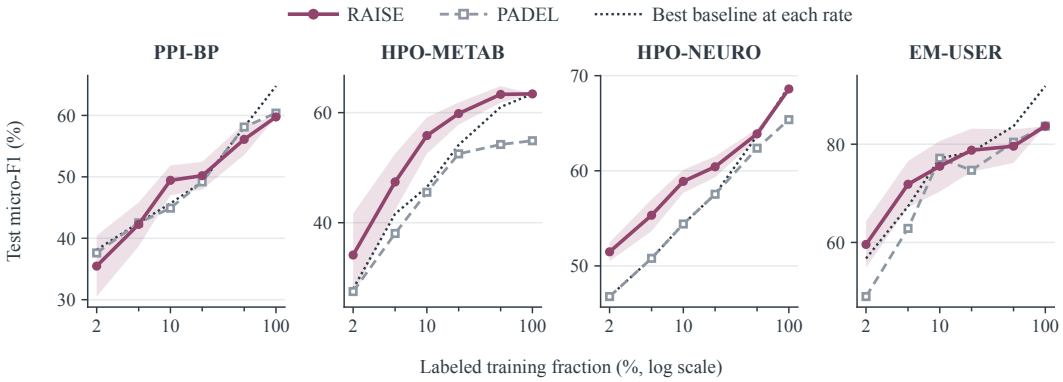


Figure C.1: Label-efficiency curves at all six rates, marked by points. Shading is RAISE's population standard deviation over the five original labeled subsets. The dotted envelope is the highest compared baseline at each rate. At 100% labels the observed subsets coincide, so the standard deviation is zero and no training pool is available for self-training.

Figure C.1 extends the label-efficiency curves to all six rates, with 100% as the fully supervised reference. Averaged over the six rates, RAISE has the highest micro-F1 and macro-F1 on every task.

Table C.1: PPI-BP: test mean ± population standard deviation across five labeled subsets (%). Boldface marks the highest mean for each metric and rate.

Method	2%	5%	10%	20%	50%	100%
Micro-F1						
Member + 1-hop Ridge	**38.11** ±2.47	40.63 ±2.13	45.66 ±1.72	48.30 ±4.81	55.72 ±1.72	61.01 ±0.00
WLKS	32.96 ±1.41	39.75 ±1.46	42.77 ±2.84	45.79 ±1.08	54.47 ±2.47	**64.78** ±0.00
SubGNN	29.06 ±3.17	33.46 ±6.36	40.88 ±1.54	45.91 ±4.59	52.20 ±2.32	56.60 ±0.00
GLASS	30.69 ±6.02	34.97 ±4.28	39.12 ±3.36	40.13 ±3.72	47.67 ±6.22	58.49 ±0.00
SSNP	35.60 ±4.11	38.24 ±2.08	40.00 ±2.24	40.88 ±2.73	40.00 ±1.41	44.03 ±0.00
S2N	32.70 ±2.07	37.86 ±4.65	42.39 ±1.72	47.17 ±3.60	57.86 ±2.29	61.89 ±1.76
CoS2N	31.95 ±3.99	37.11 ±2.73	43.27 ±1.22	44.78 ±3.31	53.33 ±2.80	58.99 ±0.25
ISNN	32.20 ±7.54	37.36 ±3.87	42.52 ±2.41	45.16 ±3.57	52.33 ±4.22	62.26 ±0.00
PADEL	37.61 ±4.93	**42.52** ±5.32	44.91 ±2.34	49.18 ±4.24	**58.11** ±2.60	60.38 ±0.00
RAISE	35.47 ±4.96	42.26 ±3.59	**49.43** ±2.44	**50.19** ±2.26	56.10 ±2.56	59.75 ±0.00
Macro-F1						
Member + 1-hop Ridge	17.43 ±1.99	23.89 ±4.52	30.65 ±1.48	35.19 ±5.00	45.91 ±3.43	52.35 ±0.00
WLKS	11.72 ±2.78	17.88 ±1.70	20.49 ±3.01	24.77 ±1.15	39.81 ±3.02	52.85 ±0.00
SubGNN	11.41 ±2.22	14.92 ±5.04	23.16 ±3.52	33.85 ±4.59	39.28 ±2.71	43.93 ±0.00
GLASS	22.73 ±3.85	27.84 ±2.74	30.07 ±3.51	32.06 ±2.89	39.41 ±6.10	49.58 ±0.00
SSNP	21.02 ±3.98	23.40 ±2.10	25.61 ±1.99	24.10 ±1.23	24.55 ±3.02	32.46 ±0.00
S2N	19.71 ±2.33	27.75 ±4.09	29.94 ±3.07	35.93 ±6.00	48.81 ±3.03	52.53 ±1.94
CoS2N	22.21 ±3.22	28.79 ±4.85	32.74 ±0.77	35.54 ±2.32	44.91 ±3.80	48.10 ±0.68
ISNN	19.20 ±7.31	23.33 ±2.95	29.36 ±1.79	32.84 ±1.73	41.06 ±4.73	53.78 ±0.00
PADEL	24.29 ±7.79	31.71 ±5.28	31.70 ±2.93	39.36 ±4.41	44.95 ±2.62	46.21 ±0.00
RAISE	**31.29** ±4.52	**37.90** ±4.00	**45.25** ±2.13	**46.14** ±2.67	**51.47** ±2.44	**54.87** ±0.00

Table C.2: HPO-METAB: test mean ± population standard deviation across five labeled subsets (%). Boldface marks the highest mean for each metric and rate.

Method	2%	5%	10%	20%	50%	100%
Micro-F1						
Member + 1-hop Ridge	21.45 ±3.47	33.11 ±4.25	40.26 ±2.48	46.81 ±5.06	53.79 ±3.61	56.17 ±0.00
WLKS	20.17 ±2.70	31.91 ±5.18	37.02 ±2.44	44.85 ±3.38	52.34 ±2.00	57.87 ±0.00
SubGNN	17.02 ±5.25	19.57 ±6.41	36.77 ±5.60	47.15 ±4.03	47.57 ±2.41	48.09 ±0.00
GLASS	23.49 ±7.16	25.28 ±2.98	29.62 ±4.45	36.43 ±7.22	58.72 ±2.98	60.43 ±0.00
SSNP	12.43 ±5.06	19.32 ±3.15	21.36 ±3.41	27.40 ±4.57	56.17 ±2.90	60.00 ±0.00
S2N	22.64 ±2.43	36.34 ±5.30	42.21 ±2.57	52.09 ±3.05	61.02 ±2.50	**63.40** ±0.00
CoS2N	28.00 ±4.45	41.62 ±6.12	46.47 ±3.61	54.13 ±6.72	58.81 ±3.01	62.72 ±2.01
ISNN	21.62 ±5.14	38.72 ±3.73	42.64 ±2.80	48.43 ±4.13	53.96 ±2.78	57.45 ±0.00
PADEL	27.49 ±3.48	38.04 ±4.00	45.53 ±3.63	52.51 ±4.22	54.21 ±2.32	54.89 ±0.00
RAISE	**34.13** ±7.55	**47.40** ±5.14	**55.83** ±3.30	**59.83** ±2.02	**63.32** ±1.51	**63.40** ±0.00
Macro-F1						
Member + 1-hop Ridge	19.63 ±4.80	31.22 ±4.18	38.45 ±2.64	45.97 ±5.00	52.13 ±3.18	54.64 ±0.00
WLKS	19.56 ±2.77	31.33 ±4.89	35.72 ±1.92	44.42 ±3.49	51.28 ±1.72	56.68 ±0.00
SubGNN	16.12 ±5.21	17.81 ±5.61	35.33 ±4.86	45.83 ±3.94	45.70 ±3.15	46.93 ±0.00
GLASS	15.81 ±4.63	18.28 ±2.89	21.10 ±4.37	32.87 ±4.54	56.94 ±3.08	57.03 ±0.00
SSNP	10.34 ±4.59	13.46 ±1.84	14.45 ±1.57	20.47 ±3.62	51.35 ±3.17	57.45 ±0.00
S2N	21.14 ±3.07	34.68 ±5.82	41.09 ±3.02	50.42 ±2.53	60.64 ±2.31	**62.85** ±0.00
CoS2N	26.79 ±6.04	37.77 ±5.74	44.01 ±5.09	53.47 ±5.43	58.63 ±2.95	62.57 ±1.45
ISNN	20.95 ±3.82	35.65 ±2.52	41.10 ±3.31	46.98 ±3.51	52.15 ±2.78	56.01 ±0.00
PADEL	27.09 ±3.50	34.27 ±3.02	43.60 ±3.93	52.53 ±4.25	52.72 ±2.97	53.93 ±0.00
RAISE	**32.28** ±6.40	**44.20** ±5.75	**54.01** ±3.62	**57.73** ±1.68	**60.78** ±1.52	60.60 ±0.00

Table C.3: HPO-NEURO: test mean ± population standard deviation across five labeled subsets (%). Boldface marks the highest mean for each metric and rate.

Method	2%	5%	10%	20%	50%	100%
Micro-F1						
Member + 1-hop Ridge	45.44 ±0.20	48.45 ±1.28	53.35 ±1.05	57.18 ±1.25	62.75 ±1.19	64.94 ±0.00
WLKS	44.56 ±0.96	47.53 ±1.65	50.14 ±1.67	52.45 ±1.91	58.82 ±0.98	64.80 ±0.00
SubGNN	39.36 ±1.02	44.17 ±1.50	50.19 ±1.87	54.67 ±0.76	59.24 ±0.54	65.00 ±0.00
GLASS	40.54 ±1.32	38.29 ±1.48	49.08 ±3.61	54.01 ±1.50	61.43 ±0.67	67.61 ±0.00
SSNP	37.34 ±2.76	35.32 ±0.47	47.28 ±0.66	53.27 ±0.99	60.71 ±0.29	67.30 ±0.00
S2N	44.62 ±1.50	49.25 ±1.06	51.80 ±1.45	55.49 ±1.13	60.84 ±0.93	66.79 ±0.12
CoS2N	44.74 ±1.56	47.83 ±1.31	52.00 ±1.64	56.09 ±1.06	61.52 ±0.40	66.55 ±0.22
ISNN	45.13 ±0.68	49.55 ±1.20	54.36 ±1.47	57.30 ±1.86	63.69 ±0.97	**68.91** ±0.00
PADEL	46.78 ±1.70	50.80 ±1.49	54.41 ±1.37	57.53 ±0.99	62.37 ±0.98	65.38 ±0.00
RAISE	**51.48** ±1.00	**55.33** ±1.67	**58.89** ±1.21	**60.42** ±1.10	**63.88** ±0.55	68.60 ±0.00
Macro-F1						
Member + 1-hop Ridge	41.94 ±0.14	45.41 ±1.01	50.52 ±0.96	54.68 ±1.20	60.82 ±1.17	63.23 ±0.00
WLKS	41.28 ±1.01	44.67 ±1.46	47.79 ±1.29	50.07 ±1.91	57.25 ±0.71	62.76 ±0.00
SubGNN	36.87 ±0.76	41.51 ±0.83	47.85 ±1.69	52.79 ±0.75	57.57 ±0.47	63.19 ±0.00
GLASS	38.22 ±1.75	35.81 ±1.54	46.28 ±3.16	51.58 ±1.29	60.02 ±0.68	66.06 ±0.00
SSNP	35.36 ±2.72	33.33 ±1.01	44.58 ±0.64	51.39 ±0.73	59.35 ±0.32	65.46 ±0.00
S2N	41.26 ±1.49	46.27 ±0.69	48.73 ±1.74	52.76 ±1.06	59.36 ±0.86	65.93 ±0.16
CoS2N	41.51 ±1.27	45.37 ±0.87	49.28 ±1.78	54.07 ±0.94	60.19 ±0.38	64.93 ±0.30
ISNN	41.87 ±0.61	46.59 ±1.18	52.14 ±1.25	55.35 ±1.74	**61.94** ±0.98	**67.38** ±0.00
PADEL	43.08 ±1.80	47.99 ±1.20	51.84 ±1.20	55.50 ±1.18	60.98 ±0.96	63.69 ±0.00
RAISE	**48.28** ±1.31	**52.22** ±1.55	**55.86** ±0.92	**57.60** ±0.96	61.66 ±0.47	66.60 ±0.00

Table C.4: EM-USER: test mean ± population standard deviation across five labeled subsets (%). Boldface marks the highest mean for each metric and rate.

Method	2%	5%	10%	20%	50%	100%
Micro-F1						
Member + 1-hop Ridge	43.67 ±3.05	67.35 ±14.55	73.88 ±7.57	76.73 ±3.05	76.73 ±5.41	81.63 ±0.00
WLKS	47.35 ±6.51	47.76 ±2.77	70.61 ±6.14	75.10 ±9.25	82.45 ±7.37	**91.84** ±0.00
SubGNN	42.45 ±2.38	42.45 ±3.96	56.73 ±14.40	55.51 ±7.57	69.39 ±3.41	70.20 ±2.08
GLASS	50.20 ±7.48	38.78 ±2.24	45.71 ±7.37	60.82 ±11.43	73.47 ±8.46	87.76 ±0.00
SSNP	51.02 ±5.77	56.33 ±8.52	56.73 ±12.27	72.24 ±5.86	81.63 ±4.28	85.71 ±0.00
S2N	44.49 ±3.51	42.86 ±4.08	50.61 ±11.57	66.94 ±10.20	70.61 ±13.14	81.22 ±2.00
CoS2N	51.84 ±7.81	44.49 ±1.53	60.82 ±7.23	65.71 ±5.97	79.59 ±6.83	87.76 ±0.00
ISNN	56.73 ±11.36	59.18 ±3.87	72.24 ±7.14	78.37 ±8.22	**83.67** ±4.83	85.71 ±0.00
PADEL	48.98 ±7.07	62.86 ±8.10	**77.14** ±2.38	74.69 ±1.63	80.41 ±4.20	83.67 ±0.00
RAISE	**59.59** ±4.72	**71.84** ±4.72	75.51 ±5.16	**78.78** ±4.40	79.59 ±3.41	83.67 ±0.00
Macro-F1						
Member + 1-hop Ridge	34.94 ±4.62	64.57 ±18.62	73.55 ±7.87	76.59 ±3.08	76.59 ±5.55	81.63 ±0.00
WLKS	39.24 ±10.12	44.96 ±3.37	69.95 ±5.62	74.67 ±9.09	82.11 ±7.52	**91.67** ±0.00
SubGNN	34.58 ±5.49	41.31 ±4.10	55.12 ±13.89	50.59 ±7.13	67.41 ±3.83	67.95 ±3.98
GLASS	45.28 ±9.00	31.07 ±2.70	40.85 ±9.11	60.06 ±10.95	73.34 ±8.38	87.50 ±0.00
SSNP	47.33 ±2.35	46.68 ±11.85	55.03 ±12.30	70.52 ±6.98	80.76 ±4.68	85.32 ±0.00
S2N	42.00 ±3.75	36.46 ±7.91	47.13 ±14.64	65.76 ±10.45	69.96 ±13.44	80.35 ±2.25
CoS2N	47.65 ±8.94	40.46 ±2.33	59.79 ±8.70	65.32 ±6.48	79.45 ±6.82	87.63 ±0.00
ISNN	47.81 ±14.70	50.35 ±2.69	67.89 ±9.82	76.82 ±8.27	**82.65** ±5.33	85.50 ±0.00
PADEL	43.41 ±11.44	62.08 ±7.98	**76.80** ±2.39	73.87 ±1.63	80.17 ±4.15	83.67 ±0.00
RAISE	**58.75** ±5.71	**71.66** ±4.71	75.25 ±5.13	**78.55** ±4.23	79.50 ±3.45	83.61 ±0.00

D LABEL-SUBSET STABILITY AND MATCHED ATTRIBUTION

The analyses below assess labeled-subset variation on fixed task graphs and effective splits. Real-task comparisons use the five original subsets (seeds 0–4) unless fresh subsets (seeds 1000–1049) are specified, and methods share the same sampled training instances within each comparison. The fresh-subset procedures are specified below; Appendix E.3 gives the five-subset paired percentile intervals.

D.1 ORIGINAL FIVE SUBSETS: EXTERNAL METHODS

The agreement analysis reuses the saved predictions underlying Table 1: four tasks, four label rates, five labeled subsets, and ten methods, giving 800 prediction arrays. At each task and rate, we average Cohen's κ over the ten pairs of labeled subsets; for multilabel predictions, binary per-label agreement is averaged over the canonical label space. Two identical constant predictions have undefined chance-corrected agreement and receive zero, which affects two of the 1,600 comparisons. We also report prediction disagreement, but near-constant predictions can obtain low disagreement despite poor F1. Table D.1 reports the task-averaged κ.

Table D.1: Chance-corrected agreement (Cohen's κ) on the original five labeled subsets. Values first average subset pairs within each task–rate cell, then tasks. Overall also averages the four rates.

Method	2%	5%	10%	20%	Overall
Member + 1-hop Ridge	0.178	0.281	0.407	0.504	0.342
WLKS	0.125	0.238	0.346	0.434	0.286
SubGNN	0.162	0.324	0.327	0.407	0.305
GLASS	0.188	0.383	0.334	0.397	0.326
SSNP	0.242	0.309	0.442	0.527	0.380
S2N	0.226	0.288	0.386	0.424	0.331
CoS2N	0.250	0.349	0.412	0.504	0.379
ISNN	0.146	0.229	0.340	0.487	0.301
PADEL	0.153	0.260	0.400	0.454	0.317
RAISE	0.231	0.474	0.538	0.635	0.469

RAISE exceeds PADEL and ISNN in all 16 task–rate cells; it exceeds Member + 1-hop Ridge, CoS2N, S2N, and SubGNN in 15 cells, WLKS and GLASS in 14, and SSNP in 12. At 2%, CoS2N and SSNP have higher task-averaged κ. Relative to PADEL, RAISE has lower disagreement in 14 cells and higher score-ranking stability in all 16.

D.2 FIFTY FRESH LABELED SUBSETS

The fresh-subset study draws 50 labeled subsets per budget with the original stratified sampling rule, and each predictor retains its geometry and method settings. Single-label sampling allocates counts in proportion to training-class frequencies by largest remainder, with at least one example per class, and samples without replacement within each class. For HPO-NEURO, greedy iterative stratification fills deficits for rare labels first, with randomized tie-breaking.

We compare WL-KRR, $\mathbf{K}_0$-KRR, and complete RAISE over four tasks at 2/5/10/20% labels, giving 2,400 conditions. Table D.2 reports prediction disagreement at each label rate; the common-regularization study in Appendix D.3 compares components under shared settings.

Table D.2: Prediction disagreement under 50 fresh labeled subsets (seeds 1000–1049). Lower is better. Each entry averages tasks and within-condition subset pairs.

Predictor	2%	5%	10%	20%
WL-KRR	0.563	0.513	0.451	0.367
$\mathbf{K}_0$-KRR	0.536	0.454	0.389	0.303
RAISE	0.490	0.393	0.320	0.246

Across the fixed tasks and rates, RAISE improves micro-F1 over WL-KRR by 7.08 points (paired labeled-subset bootstrap interval $[6.72, 7.45]$) and over $\mathbf{K}_0$-KRR by 2.05 points (interval $[1.84, 2.25]$); the corresponding macro-F1 gains are 7.09 and 2.03 points. All 50 task-and-rate-averaged subset-block differences are positive for both micro-F1 comparisons.

For these F1 intervals and the matched-attribution intervals below, we first average the paired predictor differences across the included tasks and budgets within each subset seed, so that each seed is a paired block. We then resample the 50 seed means with replacement 5,000 times and take the 2.5th and 97.5th percentiles of the resampled means as the 95% interval.

Disagreement averages over 1,225 pairs sharing the same 50 runs within each task–budget cell, so the intervals in Figure 5(a,b) use a delete-one-subset jackknife (Efron, 1982): each replicate removes the same seed from every task, budget, and predictor, and the 95% interval is the estimate $\pm\, t_{0.975,49}$ standard errors. The paired reductions relative to $\mathbf{K}_0$-KRR are 4.52 $[3.88, 5.16]$, 6.16 $[5.40, 6.92]$, 6.94 $[6.30, 7.58]$, and 5.69 $[5.21, 6.16]$ points at 2/5/10/20% labels. By affinity quartile, they are 7.26 $[6.74, 7.78]$, 6.54 $[5.96, 7.13]$, 4.88 $[4.39, 5.37]$, and 4.54 $[4.03, 5.05]$. A seed-resampling bootstrap that excludes self-pairs agrees within 0.1 points.

D.3 Matched Geometry, Self-Training, and Score Construction

The attribution study uses the same 50 fresh subsets at 5/10% labels, four tasks, and six variants (2,400 predictions). It fixes $\alpha = 0.1$, $q = 0.3$, column standardization, the prediction cohort, and the task prediction rule. WL and $\mathbf{K}_0$ each receive a direct fit and one round of self-training with independent pseudo-target residual weighting. The next variant adds diffusion aggregation to the initial prediction, the pseudo-source selection, and the final prediction. The complete variant changes only the working dependence treatment, on exactly the same pseudo sources and targets as the aggregation variant. The aggregation contrast is therefore a whole-pipeline intervention, whereas the last contrast holds the selected pseudo sources fixed.

Table D.3: Common-regularization attribution on four tasks, 5/10% labels, and 50 fresh labeled subsets (seeds 1000–1049). F1 is in percent. Disagr. is prediction disagreement; Rank corr. is score-ranking Spearman agreement. ST denotes self-training; agg. denotes class-wise diffusion aggregation.

Variant	Micro-F1	Macro-F1	κ	Disagr.	Rank corr.
WL-KRR	47.86	45.53	0.260	0.489	0.369
WL-KRR + ST	49.15	46.64	0.267	0.476	0.361
$\mathbf{K}_0$-KRR	53.54	51.15	0.359	0.430	0.495
$\mathbf{K}_0$-KRR + ST	55.39	52.96	0.397	0.397	0.527
$\mathbf{K}_0$-KRR + ST + agg.	55.74	53.34	0.462	0.356	0.623
Complete RAISE	55.94	53.52	0.462	0.356	0.625

Replacing WL by $\mathbf{K}_0$ increases κ by 0.099 before self-training and 0.130 after it; the paired micro-F1 gains are $5.68\,[5.19, 6.17]$ and $6.25\,[5.67, 6.83]$ percentage points, positive in all eight task–rate cells and all 50 aggregated subset blocks. Self-training on $\mathbf{K}_0$ adds $1.85\,[1.52, 2.20]$ points and 0.038 κ, with a κ increase in six cells. Diffusion aggregation adds $0.35\,[0.04, 0.65]$ points and 0.065 κ (seven cells), and the dependence treatment adds $0.20\,[0.13, 0.27]$ points with κ unchanged at three decimals (Table D.3).

Table D.4 breaks these sequences down by task. The kernel contrast improves both quantities on all four tasks; the diffusion aggregation gain is concentrated on the biological tasks (EM-USER decreases), and self-training does not increase HPO-NEURO agreement in this comparison. Appendix E.5 examines pseudo-label error association and sensitivity to replicated sources.

D.4 Continuous Scores and Query Support

Agreement can also be measured on continuous score rankings. On the full fresh-subset study, the average pairwise score-ranking Spearman agreement is 0.401 for WL-KRR, 0.512 for $\mathbf{K}_0$-KRR, and 0.605 for RAISE, with this ordering in all 16 task–rate cells; constant score vectors yield undefined correlations and are excluded.

Table D.4: Task dependence at 5/10% labels over 50 fresh labeled subsets (seeds 1000–1049). Columns follow the cumulative sequence WL-KRR, $\mathbf{K}_0$-KRR, $\mathbf{K}_0$-KRR + ST, $\mathbf{K}_0$-KRR + ST + agg., and complete RAISE.

Task	WL-KRR	$\mathbf{K}_0$-KRR	+ ST	+ agg.	RAISE
Cohen's κ					
PPI-BP	0.160	0.247	0.313	0.379	0.379
HPO-METAB	0.218	0.297	0.381	0.468	0.462
HPO-NEURO	0.419	0.536	0.499	0.606	0.615
EM-USER	0.245	0.358	0.395	0.393	0.393
Micro-F1 (%)					
PPI-BP	36.30	40.57	43.07	44.55	44.62
HPO-METAB	38.16	44.33	49.34	51.31	51.49
HPO-NEURO	48.99	55.39	55.05	56.53	57.10
EM-USER	68.00	73.88	74.10	70.57	70.55

For each query and labeled subset, define $s(i;\mathcal{L}) = \max_{j\in\mathcal{L}} K_{0,ij}/\sqrt{K_{0,ii}K_{0,jj}}$, the maximum affinity to a labeled instance. We average this value over the 50 subsets and group queries into affinity quartiles, using neither their class labels nor prediction correctness. Table D.5 reports disagreement within these fixed groups and thus describes the association between query affinity and cross-subset agreement.

Table D.5: Disagreement by maximum-affinity quartile over 50 fresh labeled subsets (seeds 1000–1049). Grouping does not use label values. The last column subtracts RAISE disagreement from $\mathbf{K}_0$-KRR within each group.

Affinity	WL-KRR	$\mathbf{K}_0$-KRR	RAISE	$\mathbf{K}_0$-KRR − RAISE
Weakest	0.500	0.445	0.372	0.073
Q2	0.486	0.425	0.360	0.065
Q3	0.470	0.425	0.377	0.049
Strongest	0.435	0.385	0.339	0.045

E Additional Component and Repetition Results

This appendix gives the settings and full results of the matched controls behind Table 2(a), followed by prediction-rule, working-error, repetition, error-association, and neighborhood-support diagnostics.

E.1 Direct Relation and Prediction Controls

The controls in Table 2(a) use four tasks, 2/5/10/20% training labels, and the original five labeled subsets, giving 80 conditions. Uniform diffusion replaces $\mathbf{B}$ in Equation 2 with row-normalized binary membership, while retaining the same $\mathbf{P}$. Weighted diffusion uses $\mathbf{B}$ from Equation A.1. Both responses, $\mathbf{K}_0$, and the WL kernel receive the same trace normalization. The WL depth, view mixture, and histogram settings are those specified for Table E.3 in Appendix E.2. For each kernel, class-mean prediction averages a query's similarities to the observed sources of each class. Multilabel tasks subtract negative-source means, with constant label columns set to zero. KRR uses the intercept and targets in Section 4. Every rule uses column standardization and the task prediction rule on the same $\mathcal{Z}$.

Each kernel supplies a KRR initial predictor and a self-training refit with diffusion aggregation weight zero, independent pseudo-target residual weighting, and $q = 0.3$. Each initial predictor selects its own pseudo sources by the same confidence and quota policy. For each kernel and self-training setting, one global α is chosen by mean validation micro-F1 over the 80 conditions from $\{0.003, 0.01, 0.03, 0.1, 0.3, 1, 3\}$, with candidate-order tie breaking. Class-mean prediction has no additional parameter. Table E.1 reports the resulting parameters and both test metrics, and its RAISE row repeats the reference from Table 1.

Table E.1: Complete scores and parameter values for Table 2(a). Entries are mean F1 (%) over the listed label rates. ST denotes self-training.

			2/5/10/20%		5/10/20%	
Geometry	Predictor	α	Micro-F1	Macro-F1	Micro-F1	Macro-F1
WL	Class mean	–	46.46	44.80	48.74	47.08
WL	KRR	0.3	48.35	46.03	51.05	48.88
WL	KRR + ST	0.3	50.13	47.78	53.05	50.64
Uniform diffusion	Class mean	–	45.59	43.30	47.68	45.43
Uniform diffusion	KRR	0.1	53.24	50.69	56.61	53.96
Uniform diffusion	KRR + ST	0.1	53.84	51.39	57.27	54.67
Weighted diffusion	Class mean	–	50.72	47.97	53.25	50.32
Weighted diffusion	KRR	0.3	54.19	51.55	57.45	54.76
Weighted diffusion	KRR + ST	0.3	54.73	52.08	58.04	55.30
$\mathbf{K}_0$	Class mean	–	50.94	48.66	54.04	51.63
$\mathbf{K}_0$	KRR	1	53.57	51.16	57.10	54.62
$\mathbf{K}_0$	KRR + ST	0.3	55.12	52.63	58.70	56.21
RAISE	Full procedure	0.1	55.40	52.94	58.81	56.36

Fixed-α comparisons set every KRR rule to 0.1 and retain the same source-selection procedure. At the parameters in Table E.1, the five-seed conditional intervals are $[0.81, 0.98]$ for weighted minus uniform diffusion after self-training, $[3.67, 5.37]$ for weighted diffusion minus WL after self-training, and $[2.00, 3.25]$ for KRR minus class-mean prediction on $\mathbf{K}_0$ before self-training. At fixed $\alpha = 0.1$, the corresponding intervals are $[0.79, 1.18]$, $[5.07, 6.31]$, and $[1.54, 3.22]$. These intervals use the same exhaustive resampling of five paired seed means described in Appendix E.3.

At fixed $\alpha = 0.1$, one round of self-training improves uniform diffusion, weighted diffusion, WL, and $\mathbf{K}_0$ by 0.60, 0.70, 0.94, and 1.56 micro-F1 points. Weighted minus uniform diffusion is $+0.90$ before self-training and $+1.00$ after it. The corresponding differences at the Table E.1 parameters are $+0.95$ and $+0.89$. In the selected self-training fits, weighting changes the four task means by $+2.36$, $+3.51$, $+0.96$, and -3.27 in PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER order. Fixed-α KRR exceeds class-mean prediction by 3.42 points on weighted diffusion and 2.41 on $\mathbf{K}_0$. All five seed means are positive for both weighting comparisons after self-training and for both KRR-versus-class-mean comparisons. Weighted and uniform diffusion after self-training exceed WL by 5.77 and 4.77 points at fixed α, with positive mean differences for all five seeds.

E.2 Geometry, Prediction-Rule, and Working-Error Controls

Table E.2 complements the prediction-rule comparison in Table E.3 with fixed-parameter geometry controls. Both use the same 60 low-label conditions and column standardization, with no diffusion aggregation or self-training. The WL input in both tables is $\mathbf{K}_{\mathrm{WL}} = \mathrm{TN}(\beta_{\mathrm{WL}}\mathbf{H}_{\mathrm{bg}}\mathbf{H}_{\mathrm{bg}}^{\top} + (1-\beta_{\mathrm{WL}})\mathbf{H}_{\mathrm{int}}\mathbf{H}_{\mathrm{int}}^{\top})$: $\mathbf{H}_{\mathrm{bg}}$ pools WL colors from the background graph, while $\mathbf{H}_{\mathrm{int}}$ pools colors computed separately on each induced subgraph, both from uniform initial colors. In PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER order, the depths are 2/2/3/2 and β_{WL} is 0.99/0.999/0.999/0.9. Histogram normalization is enabled for the first two tasks, and only the final-depth histograms are used. These are the published WLKS view settings; the external WLKS baseline in Table 1 searches depths up to these maxima, whereas the controls fix the listed depth and trace-normalize the mixed Gram matrix. With the same KRR parameter $\alpha = 0.3$, $\mathbf{K}_0$ improves over the tested WL kernel by 5.97 points (53 wins, three ties, four losses) and over the membership kernel by 1.87 points. Diffusion alone exceeds $\mathbf{K}_0$ by 0.44 points in this fixed-α control (Table E.2), whereas $\mathbf{K}_0$ leads after self-training (Table 2a); the membership contribution varies by task. With independently tuned prediction rules, KRR exceeds SVC by 2.81 points on $\mathbf{K}_0$, with positive task-level means on all four datasets, and by 0.38 points on the WL kernel.

Table E.2: Geometry and prediction controls on 60 low-label conditions. All use column standardization, no class-wise diffusion aggregation, and no self-training. Each row subtracts the second predictor from the first. Kernel comparisons fix KRR $\alpha = 0.3$.

Paired difference	Micro-F1 (pp)	Win/tie/loss
$\mathbf{K}_0 - \mathbf{K}_{\mathrm{WL}}$	$+5.97$	53/3/4
$\mathbf{K}_0 - \mathbf{K}_{\mathrm{mem}}$	$+1.87$	47/3/10
$\mathrm{TN}(\mathbf{R}_2) - \mathbf{K}_0$	$+0.44$	36/7/17
KRR $-$ SVC on $\mathbf{K}_0$	$+2.81$	42/6/12
KRR $-$ SVC on $\mathbf{K}_{\mathrm{WL}}$	$+0.38$	32/6/22

Table E.3: Propagation and score normalization at 5/10/20% labels. Entries are mean Micro-F1 (%), without diffusion aggregation or self-training. LGC retains native normalization. The last column is the $\mathbf{K}_0$ entry minus the WL entry, in percentage points.

Predictor	Scores	WL	$\mathbf{K}_0$	$\mathbf{K}_0-$WL
Probability LGC	Raw	44.49	41.47	-3.02
	Column-standardized	50.63	54.41	$+3.78$
KRR	Raw	51.80	55.61	$+3.81$
	Column-standardized	51.41	57.10	$+5.68$

Column standardization has task-dependent effects: on the response kernel at $\alpha = 0.3$, it changes micro-F1 by -2.60, $+1.22$, approximately $+0.01$, and $+9.80$ points on PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER, respectively.

The propagation arms in Table E.3 implement LGC on the same input kernel as the ridge arms. Let $\mathbf{P}_{\mathrm{lgc}}$ denote the kernel after zeroing its diagonal and symmetric degree normalization, and let $\mathbf{Y}_{\mathrm{seed}}$ contain targets on $\mathcal{L}$ and zeros elsewhere. We consider $\mu \in \{0.0001, 0.001, 0.01, 0.1, 1, 10, 100\}$, corresponding to propagation weight $1/(1+\mu)$, and the one-step endpoint $\mathbf{P}_{\mathrm{lgc}}\mathbf{Y}_{\mathrm{seed}}$ on rows outside $\mathcal{L}$. Scores are normalized to class probabilities, across classes in single-label tasks and within each positive/negative problem in HPO-NEURO. The probabilities match scikit-learn's LabelSpreading in the corresponding single-label and independent binary multilabel cases. Column-standardized LGC selects $\mu = 10$ for $\mathbf{K}_0$ and $\mu = 0.1$ for WL, whereas raw LGC selects the one-step endpoint for $\mathbf{K}_0$ and $\mu = 100$ for WL. Column-standardized KRR selects $\alpha = 1$ for both kernels, while raw KRR selects $\alpha = 0.1$ for $\mathbf{K}_0$ and $\alpha = 0.3$ for WL. RAISE retains its separate final $\alpha = 0.1$. The eight entries in Table E.3 use these independently selected configurations. Under column standardization, the $\mathbf{K}_0$-versus-WL win/tie/loss counts over 60 conditions are 47/5/8 for LGC and 55/1/4 for KRR. KRR versus LGC on $\mathbf{K}_0$ gives 40/5/15.

Table E.4 compares working residual covariances at the final $\alpha = 0.1$. Independent treatment uses $\mathbf{I}$ on the pseudo block, and diagonal treatment uses $2\mathbf{I}$. Member correlation uses $\mathbf{I} + \bar{\mathbf{M}}_{\mathcal{P}}\bar{\mathbf{M}}_{\mathcal{P}}^{\top}$, target correlation uses $\mathbf{I} + \bar{\mathbf{T}}_{\mathcal{P}}\bar{\mathbf{T}}_{\mathcal{P}}^{\top}$, and RAISE uses Equation 7. The shuffled-member control permutes the member Gram correspondence while retaining its diagonal and spectrum. All self-training arms have the same sources and targets.

E.3 Matched Pipeline Refinements

Table E.5 lists the fixed-parameter pipelines with both metrics. Each self-training pipeline uses four tasks, four main-text rates, and five original labeled subsets, yielding 80 conditions, and all use the same $\alpha = 0.1$. Removing the diffusion aggregation changes the initial prediction and the refit and regenerates the pseudo sources, whereas within each aggregation setting the independent and dependence-aware refits share identical sources and pseudo-targets. The diffusion-only pipeline substitutes $\mathrm{TN}(\mathbf{R}_2)$ for $\mathbf{K}_0$ and uses no diffusion aggregation or dependence modeling. Over 5/10/20% labels, the initial, independently refitted, and dependence-aware predictors reach 57.53, 58.56, and 58.81 micro-F1. Their increments, computed before rounding, are 1.04 and 0.24 points. The diffusion aggregation gains in the final dependence-aware pipeline over the four main budgets are $+1.45$, $+1.74$, $+1.44$, and -2.86 on PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER.

Table E.4: Working-dependence comparisons with the same $\alpha = 0.1$, geometry, initial prediction, and sources. Diagonal and dependence models have pseudo-block diagonal two, whereas independent residual weighting has diagonal one. Entries are mean F1 (%) over the listed label rates.

	All six rates		5/10/20%	
Procedure	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Without self-training	58.41	55.99	57.53	55.02
Independent residual weighting	59.19	56.75	58.56	56.12
Diagonal working dependence	59.26	56.82	58.63	56.19
Shuffled member correlation	59.24	56.81	58.60	56.18
Member correlation	59.34	56.88	58.82	56.33
Target correlation	58.81	56.39	58.12	55.69
Member × target (RAISE)	59.36	56.92	58.81	56.36

Table E.5: Fitting stages at fixed $\alpha = 0.1$: mean F1 (%) over the listed label rates. Agg. denotes class-wise diffusion aggregation and applies to both the initial prediction and the refit. Pipelines that differ only in the dependence model share pseudo-labeled examples and targets. Bold marks the best value in each column.

				2/5/10/20%		5/10/20%	
Kernel	Agg.	ST rounds	Dependence	Micro-F1	Macro-F1	Micro-F1	Macro-F1
$\mathbf{K}_0$	Yes	0	–	53.99	51.55	57.53	55.02
$\mathrm{TN}(\mathbf{R}_2)$	No	1	No	54.84	52.36	58.03	55.51
$\mathbf{K}_0$	No	1	No	54.92	52.50	58.34	55.92
$\mathbf{K}_0$	Yes	1	No	55.14	52.69	58.56	56.12
$\mathbf{K}_0$	No	1	Yes	54.96	52.52	58.36	55.92
$\mathbf{K}_0$ (RAISE)	Yes	1	Yes	**55.40**	**52.94**	**58.81**	**56.36**

For the external paired interval, we first subtract PADEL from RAISE within every task, rate, and subset condition. We then average within each of the five subset seeds over the four tasks and four main-text rates. Enumerating the 5^5 equally likely length-five resamples of these seed means gives the percentile bootstrap intervals (Efron, 1982) $[4.47, 5.12]$ for micro-F1 and $[6.22, 7.23]$ for macro-F1, in percentage points. At the condition level, RAISE's micro-F1 exceeds PADEL's in 61 of the 80 task–rate–subset conditions, with four ties.

The same construction gives a main-range micro-F1 interval of $[0.083, 1.056]$ points for RAISE minus the simplest diffusion pipeline, and $[-0.153, 1.037]$ for the diffusion aggregation's contribution to the final pipeline. The respective seed-level differences are positive in four and three of five cases. With 50 fresh subsets at 5/10% labels, the analogous aggregation contrast gives 0.35 $[0.04, 0.65]$ points (Appendix D.3). The full method has the highest main-range and 5/10/20% averages in both metrics.

E.4 Dependence Treatments on Fifty Fresh Subsets

At 10% labels, we draw 50 fresh subsets per task (seeds 1000–1049) with the original stratified sampling rule. The fixed-parameter independent, diagonal, RAISE, and shuffled-RAISE treatments share each subset, its initial prediction, and its pseudo sources. Table E.6 reports the 800 predictions.

RAISE improves over independent treatment by 0.149 points on average, with a 95% interval of $[0.043, 0.258]$ points from a paired bootstrap over the 50 subset seeds with 5,000 resamples. Its gains over diagonal and shuffled treatment are 0.113 and 0.101 points, and most of the gain over independent treatment comes from HPO-NEURO ($+0.580$).

The complete method's 10% mean is 58.02 on these fresh subsets (Table E.6) and 59.92 on the original five (Table G.1); the two studies use different labeled subsets.

Table E.6: Fixed-parameter repetition at 10% labels using 50 fresh labeled subsets (seeds 1000–1049) per task. Entries are test Micro-F1 (%). The final column is RAISE minus independent treatment, in percentage points to three decimals.

Task	Independent	Diagonal	Shuffled	RAISE	RAISE − Independent
PPI-BP	46.18	46.24	46.18	46.30	+0.126
HPO-METAB	54.84	54.54	54.71	54.73	−0.111
HPO-NEURO	58.11	58.49	58.48	58.69	+0.580
EM-USER	72.37	72.37	72.33	72.37	+0.000
Average	57.87	57.91	57.92	58.02	+0.149

E.5 Observed Error Association

For each pseudo-labeled example, we record an error indicator for single-label tasks or the fraction of incorrect labels for HPO-NEURO after prediction. A pair is evaluated only on repetitions where both sources were selected. We require at least ten joint selections across all 50 repetitions, or five within each half, and nonzero conditional error variance for both sources. We compare overlap-weighted error correlation with 99 permutations of member correspondence within strata defined by modal pseudo target and coarse source error rate. Table E.7 reports the excess over the permutation mean.

Table E.7: Excess overlap-weighted error correlation above within-stratum permutations, from the 50 fresh labeled subsets (seeds 1000–1049) at 10% labels. Estimates use pairs jointly selected at least ten times overall or five times within each half.

Task	All 50	First 25	Last 25	Eligible pairs	Overlapping
PPI-BP	+0.331	+0.304	+0.341	9,586	281
HPO-METAB	+0.101	+0.034	+0.111	671	312
HPO-NEURO	+0.020	+0.019	+0.019	618,416	213,259
EM-USER	−0.151	−0.040	−0.544	18	15

Overlap association is positive on PPI-BP, HPO-METAB, and HPO-NEURO in both halves; EM-USER contributes only 18 eligible pairs overall and seven in the second half. The estimates describe jointly selected sources with variable errors, and source pairs are dependent. A complementary single-fit diagnostic centers residuals within exact-pseudo-target and coarse-confidence strata, and all 60 original low-label conditions show positive excess overlap-weighted residual alignment.

Replicating pseudo observations four times in the final regression, with the diffusion aggregation and unique sources fixed, changes test scores by RMS 0.08459 for independent treatment, 0.04569 for RAISE, and 0.04371 for a shuffled dependence model. Both dependence models reduce this score drift, and count normalization eliminates it.

E.6 Label Agreement in Kernel Neighborhoods

To relate the aggregate gains to individual queries, we keep the predictions from Section 5.3 fixed and examine their labeled neighbors. For each kernel, the diagnostic averages label agreement with up to five nearest labeled subgraphs, treating zero similarity as no support. Agreement is a same-class indicator for single-label tasks and label-set Jaccard overlap for multilabel tasks. Queries are grouped by which kernel supplies higher support; grouping uses labels and kernel similarities, not predictions.

Queries with stronger $\mathbf{K}_0$ label support have larger $\mathbf{K}_0$ gains under both predictors (Figure E.1), and relative support and instance-F1 gain are positively correlated in 57 of 60 KRR conditions and 55 of 60 LGC conditions (Table E.8). These descriptive associations connect the kernel gains to the labels of nearby examples.

The diagnostic reuses the fixed $\mathbf{K}_0$/WL predictions under standardized KRR and probability LGC for the 60 conditions in Table E.3. Let $\mathcal{N}_K(i)$ contain the $k = \min(5, |\mathcal{L}|)$ nearest labeled sources under cosine-normalized kernel similarity $\bar{K}_{ij} = K_{ij}/\sqrt{K_{ii}K_{jj}}$, with ties ordered by instance identity. For positive-class sets $\mathcal{C}_i = \{c : Y_{ic} = 1\}$ and $\mathcal{C}_j = \{c : Y_{jc} = 1\}$, kernel-neighborhood label

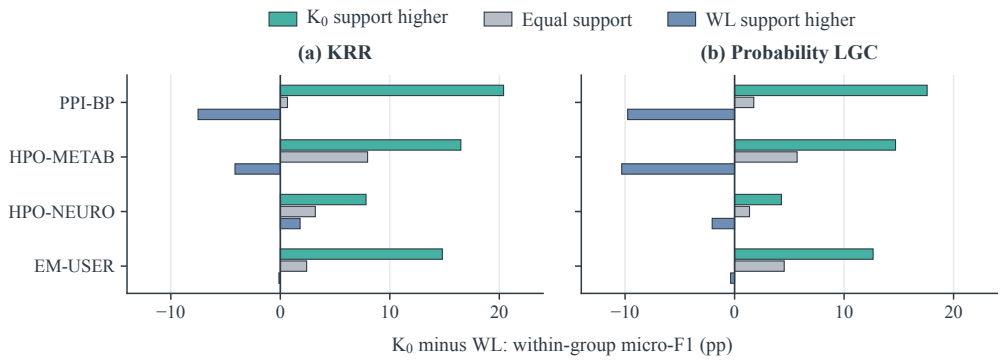


Figure E.1: Prediction gain grouped by relative label support. Bars show the within-group micro-F1 difference between $\mathbf{K}_0$ and WL, averaged over 15 conditions per task. Both predictors use column-standardized scores, and the groups do not use predictions.

agreement is defined as follows:

$$h_K(i) = \frac{1}{k} \sum_{j \in \mathcal{N}_K(i)} \mathbf{1}\{\bar{K}_{ij} > 10^{-12}\} \frac{|\mathcal{C}_i \cap \mathcal{C}_j|}{|\mathcal{C}_i \cup \mathcal{C}_j|}. \tag{E.1}$$

An empty union contributes zero. The sign of $h_{\mathbf{K}_0}(i) - h_{\mathrm{WL}}(i)$ in Equation E.1 defines the three groups in Figure E.1. We compute micro-F1 within each group, then average the 15 conditions of each task equally. Instance-F1 is used for the query-level Spearman analysis and coarse controls. It differs from pooled micro-F1 on the multilabel task.

Table E.8: Coverage of the kernel-neighborhood label-agreement diagnostic. The three groups count query visits across 15 labeled-subset conditions per task, shared by both predictors. Repeated visits are not independent instances. The final columns count positive label-agreement–gain Spearman correlations.

Task	$h_{\mathbf{K}_0} > h_{\mathrm{WL}}$	Equal	$h_{\mathbf{K}_0} < h_{\mathrm{WL}}$	KRR	LGC
PPI-BP	839	1,023	523	15/15	15/15
HPO-METAB	991	1,325	1,209	15/15	15/15
HPO-NEURO	2,926	549	2,465	15/15	15/15
EM-USER	303	182	250	12/15	10/15

Direct member overlap gives a less uniform account of the gains: on PPI-BP, queries without direct member overlap with any labeled source still benefit from $\mathbf{K}_0$, and diffusion similarities show no consistent monotonic gain across coverage terciles.

For coarse controls, we stratify within each task, rate, and seed by class, median-split subgraph size, and median-split maximum WL similarity to $\mathcal{L}$. Multilabel queries may belong to several positive-class strata. We compare groups with higher $\mathbf{K}_0$ and higher WL support only when each has at least three query visits in the stratum, weighting the difference in mean instance-F1 gain by the smaller group count. There are 40, 101, 497, and eight eligible matched strata on PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER. Controlled contrasts are positive on the first three tasks for both predictors. EM-USER is positive for KRR and negative for LGC. The two heads together account for 25,170 visits to queries reused across conditions.

F Support-Gap Queries

A support-gap query has a true class that is observed in $\mathcal{L}$ yet absent among its closest structural matches. The diagnostic uses the observed training set $\mathcal{L}$ and the fixed WL view defined in Appendix E, including both internal and background histograms. We cosine-normalize its Gram entries and rank the labeled sources for each test query. A true class lacks direct support if it occurs in $\mathcal{L}$ but no positive-similarity source of that class can occur in a valid top-k neighborhood. The main window is $k = \min(5, |\mathcal{L}|)$, and $k = \min(10, |\mathcal{L}|)$ is a fixed sensitivity check. Values at most 10^{-12} give no

support. At the boundary value t, scores within $10^{-12} + 10^{-10}|t|$ are treated as tied. We declare a gap only when the class is absent under every valid boundary choice. Ambiguous cases remain separate.

A query belongs to the gap group when at least one true class lacks support; Figure F.1(a) and the query percentage in Table F.1 use this definition. The positive-target percentage instead averages missing-class indicators within each query's true label set before averaging queries. On HPO-NEURO these are 15.73% and 10.18%, respectively. Every queried class occurs in the labeled sets, so the measured gaps concern its coverage within the local neighborhood.

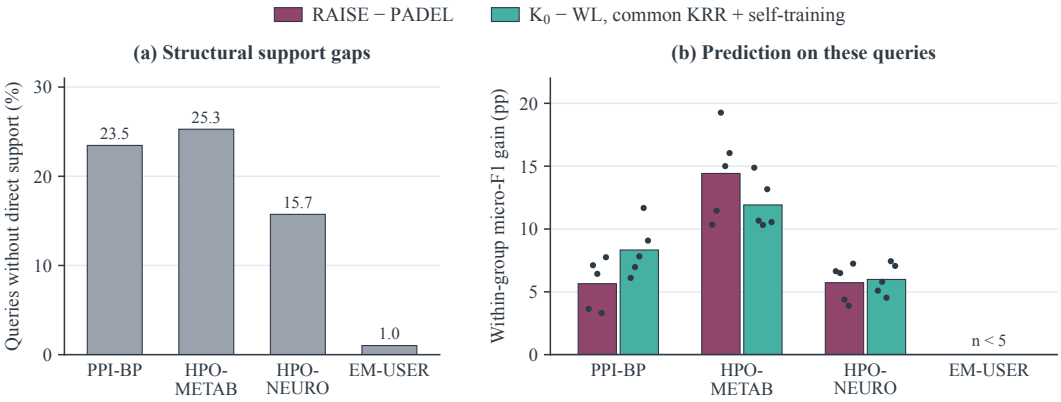


Figure F.1: Support-gap coverage and prediction gains over all support-gap queries. (a) Fraction of test queries with a true class observed in $\mathcal{L}$ but absent among their five WL-nearest labeled sources. (b) Micro-F1 gains on the same queries, averaged over four label rates and five labeled subsets. Dots denote subset-seed means. EM-USER has fewer than five such queries per condition. This population differs from the stricter within-query matched comparison in Appendix G.2.

Table F.1: Coverage of the fixed support-gap groups at $k = 5$. Distinct queries are counted once across the 20 repeated conditions. The final columns use query and positive-target denominators, respectively.

Task	Test	Gap range	Distinct	Query %	Target %
PPI-BP	159	22–59	142	23.46	23.46
HPO-METAB	235	43–82	217	25.28	25.28
HPO-NEURO	396	41–77	244	15.73	10.18
EM-USER	49	0–4	8	1.02	1.02

Within matched task, budget, subset, and true-class strata, gap targets have higher omission rates than supported targets. The differences are 15.72/38.08/35.84 points for PADEL and 3.39/36.30/43.22 for WLKS on PPI-BP, HPO-METAB, and HPO-NEURO, with at least five queries in each compared group. WL neighborhoods improve coverage relative to random selection: uniformly drawing the same number of labeled sources without replacement gives expected positive-target gap rates of 33.73%, 40.90%, and 16.95% on PPI-BP, HPO-METAB, and HPO-NEURO, exceeding the corresponding WL rates. Within the WL gaps, uniform-diffusion neighborhoods show no consistent coverage gain over this random reference.

Table F.2 reports both metrics on these queries. All group scores use the same full-prediction-set decisions, retaining the original normalization and multilabel quotas when computing group metrics. A group enters a task mean when it has at least five queries; excluded conditions retain their raw counts. The matched-kernel comparison reuses the self-training predictions of Table E.1 at the globally selected $\alpha = 0.3$ for both kernels; its initial prediction and refit use zero diffusion aggregation weight, independent residual weighting, and $q = 0.3$. Fixed-$\alpha = 0.1$ comparisons additionally cover the initial predictor. All three comparisons improve the biological task means at both windows (Table F.3).

For the unsupported positive targets themselves, we count both corrected omissions and new omissions. Their net reduction relative to PADEL is 5.65, 14.42, and 5.08 points on PPI-BP, HPO-

Table F.2: F1 (%) on support-gap queries. The full label vector is evaluated on multilabel queries. Groups with fewer than five queries are retained in Table F.4 instead of entering these means.

	Micro-F1			Macro-F1			
Task	WLKS	PADEL	RAISE	WLKS	PADEL	RAISE	Conditions
PPI-BP	18.79	19.40	25.05	5.78	14.15	22.22	20/20
HPO-METAB	5.67	10.16	24.58	5.05	9.20	21.44	20/20
HPO-NEURO	31.64	36.02	41.75	29.79	34.72	41.02	20/20
EM-USER	–	–	–	–	–	–	0/20

Table F.3: Micro-F1 gains (pp) on the same fixed query groups. The last three columns compare $\mathbf{K}_0$ with WL under common KRR, column standardization, and task prediction rule. Self-training fits use $q = 0.3$ without diffusion aggregation or dependence modeling. ST denotes self-training. The last column uses the globally selected $\alpha = 0.3$ of Table E.1 for both kernels.

			$\mathbf{K}_0$-KRR − WL-KRR		
Task	k	RAISE − PADEL	$\alpha = 0.1$	$\alpha = 0.1$, + ST	$\alpha = 0.3$, + ST
PPI-BP	5	5.65	10.91	6.69	8.33
HPO-METAB	5	14.42	7.68	10.42	11.91
HPO-NEURO	5	5.73	6.29	6.44	5.98
PPI-BP	10	10.22	12.26	3.45	6.81
HPO-METAB	10	10.43	5.56	9.78	9.99
HPO-NEURO	10	5.31	3.27	3.36	2.40

METAB, and HPO-NEURO, using equal query weight divided among each query's true classes. HPO-NEURO's full-query micro-F1 gain is 5.73 points because it also evaluates supported classes and false positives. On those queries, mean false positives decrease by 0.1102 per query and omissions by 0.1522. The five labeled-subset seed means are positive for each biological task. PPI-BP at 2% is less stable, with two wins, one tie, and two losses.

To account for differences in class composition between the gap and supported groups, we compare them within each task, budget, subset, and true class only when each contains at least five queries, then average matched classes and conditions equally. At $k = 5$, RAISE's omission reductions in the gap/supported groups are 8.61/13.54, 17.64/7.81, and 9.55/6.48 points on the three biological tasks. At $k = 10$, HPO-NEURO's smaller matched subset gives $-0.55/10.33$ points over 15 conditions and 19 class strata. This grouping differs from the relative-support diagnostic in Appendix E.6, which compares two kernels' support under KRR or LGC.

EM-USER has five nonempty $k = 5$ conditions, each containing 1–4 queries (ten visits of eight distinct queries), so Table F.4 reports the individual cases.

Table F.4: All nonempty EM-USER support-gap conditions at $k = 5$. Ten query visits involve eight distinct queries. Each condition is below the five-query threshold used for group averages.

Label rate	Seed	Queries	PADEL correct	RAISE correct
10%	1	1	1	1
10%	2	4	3	2
20%	2	1	0	0
20%	3	1	1	1
20%	4	3	1	0

G Correspondence Interventions and Their Scope

This appendix details the three interventions of Section 5.4: relation reassignment, the within-query support comparison with source exchange, and the relation-strength control.

G.1 Reassigning Instance–Relation Correspondence

At 10% labels and the original five labeled subsets, the reassignment experiment applies a symmetric instance permutation to the kernel:

$$\mathbf{K}_\rho = \mathbf{\Pi}_{\mathrm{inst},\rho}\mathbf{K}_0\mathbf{\Pi}_{\mathrm{inst},\rho}^\top.$$

The matrix $\mathbf{\Pi}_{\mathrm{inst},\rho}$ is an $N \times N$ instance permutation, and the same permutation reassigns the diffusion and membership views. The instance labels, the training/validation/test partition, the labeled-subset identities, and the prediction settings are held fixed. Unlike the joint vertex renumbering in Appendix A, which leaves the Gram unchanged, this permutation changes instance–relation correspondence. We report the realized moved fraction for each finite permutation.

The permutation preserves the full matrix's spectrum, rank, positive semidefiniteness, trace, and Frobenius norm (numerical trace and norm discrepancies are at most 5×10^{-13}), but it reassigns individual diagonal values and label relations and can change the spectrum of the source submatrix $\mathbf{K}_{0,\mathcal{L},\mathcal{L}}$. The intervention therefore measures correspondence reassignment with whole-matrix properties fixed and local source conditioning free to vary. At zero perturbation, it reproduces the original predictor. Table G.1 reports whole-task means, and Table G.2 gives the per-task differences from $\mathbf{K}_0$-KRR.

Table G.1: Whole-task means under correspondence reassignment, at 10% labels. F1 is in percent, and differences are in percentage points.

Requested fraction	0	0.02	0.05	0.10	0.20
RAISE Micro-F1	59.92	58.84	57.10	54.74	49.52
$\mathbf{K}_0$-KRR Micro-F1	57.82	57.20	55.85	53.36	47.41
RAISE − WL-KRR	8.23	7.16	5.41	3.05	−2.17
RAISE − $\mathbf{K}_0$-KRR	2.09	1.64	1.25	1.37	2.11

RAISE's mean micro-F1 decreases monotonically from 0 to 20% reassignment, while its advantage over $\mathbf{K}_0$-KRR remains positive throughout (2.11 points at 20%).

Table G.2: RAISE − $\mathbf{K}_0$-KRR Micro-F1 by task under correspondence reassignment at 10% labels, in percentage points.

Task	0	0.02	0.05	0.10	0.20
PPI-BP	2.64	2.94	1.94	1.81	2.47
HPO-METAB	8.17	7.42	6.93	6.89	6.52
HPO-NEURO	0.83	0.92	1.13	0.87	1.40
EM-USER	−3.27	−4.73	−4.98	−4.08	−1.96

G.2 Within-Query Support States and Source Exchange

The stricter within-query comparison fixes the query, the label budget, and the full training-class count vector, and retains only queries that have both definite support and a definite gap across the original labeled subsets. Predictions are averaged within a state, then within a query and budget. At $k = 5$, the gap state's additional PADEL error is 9.98 points on PPI-BP and 23.73 on HPO-METAB. The corresponding WLKS changes are 4.88 and 21.97 points. In these particular gap states, complete RAISE minus PADEL is -0.17 and 17.96 points. Appendix F reports the complementary broader gap-group analysis with its own population and weighting.

The source-exchange experiment holds the query, the number of observed labels, and their class counts fixed, and replaces one source by one of two same-class candidates. Candidate–query and candidate–common-source WL similarities are approximately matched, size and mean-degree ratios are at most 1.25, and diffusion-cosine differences are at least 0.05. Pairs are selected before prediction from these structural and relational criteria, independently of query labels and prediction scores. The two candidate sources share a class with each other, and both query-matching and query-mismatching classes are included. Both kernels use KRR with $\alpha = 0.1$, no self-training, and the same normalization and prediction rule.

On PPI-BP, these constraints admit 733 source pairs covering 156 distinct queries. Averaging first within each query, so that repeated queries receive equal weight, the exchange raises WL accuracy from 33.75% to 34.55% and $\mathbf{K}_0$ accuracy from 37.62% to 42.51%, gains of 0.80 and 4.89 points that differ by 4.09 points. Figure 4(a) shows the corresponding true-class margins for same-class and other-class swaps.

G.3 Relation Strength and Informative Structure

The controlled relation-strength test varies the relational signal while holding memberships, labels, shape, degree assignment, and splits paired; Table G.3 uses 5% labels and five graph seeds. The strengthened WL-KRR selects among final-layer and accumulated WL views at depths 1–5 by validation, while RAISE uses the same settings as in the real-task study. In these controlled tasks,

Table G.3: Micro-F1 (%) at 5% labels when internal shape and background relations carry class information to different degrees, averaged over five graph seeds.

Shape	Relation	WL-KRR	$\mathbf{K}_0$-KRR	RAISE
Informative	Strong	75.00	87.58	98.00
Informative	Intermediate	75.00	65.42	73.00
Informative	Class-independent	75.00	52.83	51.17
Class-independent	Strong	50.00	87.92	98.67
Class-independent	Intermediate	50.00	65.17	71.08
Class-independent	Class-independent	50.00	49.00	49.92

relative performance depends on where the class information resides: RAISE leads the compared predictors in the strong-relation conditions, while WL-KRR leads when shape is informative and relations are class-independent. The regular and variable-degree studies in Appendix H also show successful recovery by LGC, so informative background relations are useful inputs to both fitted and propagation-based predictors.

H Controlled Structural and Relational Information

On the real benchmarks, internal WL descriptions have conflicting-label group coverage of 56.51%, 23.50%, 24.73%, and 0% on PPI-BP, HPO-METAB, HPO-NEURO, and EM-USER. These fractions measure how often differently labeled instances share an internal descriptor. Adding the background WL view makes the combined descriptions distinct for every instance in each real task, so the controlled constructions below separate the role of exact descriptor collisions from the predictive value of relations.

The regular construction has 1,200 disjoint three-vertex sets in a connected 8-regular graph. Each set induces a path or triangle and has one of two balanced labels. A fraction $\epsilon \in \{0, 0.25, 0.5\}$ of shape assignments is reversed within each class and split. Internal WL descriptions then have two types and all background WL node colors coincide, so the combined view's classification ceiling is $1 - \epsilon$. In the informative condition, 90% of external edges join same-class sets (10% cross classes). The null condition uses $\epsilon = 0.5$ and class-independent external edges. A joint vertex permutation removes any relation to vertex numbering. Five graph seeds share paired targets, memberships, and label subsets across the conditions. Table H.1 reports the results.

The variable-degree construction retains $\epsilon = 0.5$ and the same memberships and targets, but each set's members have background degree 6, 8, or 10. Each degree occurs equally often within every class–shape combination, so mean degree remains eight and degree does not determine the class. External edges are sampled within class and rewired to 90% same-class (10% cross-class) edges in the informative condition, whereas the null condition samples them without class restrictions. Both conditions preserve the same degree assignment, observed labels, and splits. We evaluate all five predetermined graph seeds.

In the ten variable-degree graphs, depth-one combined WL descriptions form six groups, all containing conflicting labels. At depth two, 998–1,025 descriptions distinguish most instances, with mean conflicting-instance coverage of 16.45%. At depths three through five, every designated set has a

Table H.1: Regular-graph Micro-F1 (%) over five graph seeds, reported as mean ± population standard deviation. Mismatch is the fraction ϵ of reversed shape assignments.

Mismatch	Relation	Label rate	WLKS	LGC	RAISE
0.00	Signal	5%	100.00 ±0.00	94.75 ±0.68	98.58 ±0.57
0.25	Signal	5%	75.00 ±0.00	95.25 ±2.05	99.08 ±0.81
0.50	Signal	2%	50.00 ±0.00	84.42 ±10.98	96.25 ±1.75
0.50	Signal	5%	50.00 ±0.00	95.33 ±1.38	98.08 ±0.77
0.50	Signal	10%	50.00 ±0.00	96.17 ±0.41	99.00 ±0.42
0.50	Signal	20%	50.00 ±0.00	98.00 ±1.03	99.42 ±0.20
0.50	Null	5%	50.00 ±0.00	50.75 ±2.90	50.92 ±2.60

distinct description and conflict coverage is zero. The depth-five kernel classifier nevertheless remains at 50.00 in every signal and null condition (Table H.2).

In all ten graphs, the depth-five background WL kernel is the identity to numerical precision. Let $\mathcal{D}_{\mathrm{te}}$ denote test queries and $(\mathbf{H}_{\mathrm{shape}})_{ij}$ indicate equal path or triangle shape. The normalized internal kernel is $\mathbf{H}_{\mathrm{shape}}$, so the WL kernel and its query–source block take the following form:

$$\mathbf{K}_{\mathrm{WL}} = 0.9\mathbf{I}_N + 0.1\mathbf{H}_{\mathrm{shape}}, \qquad (\mathbf{K}_{\mathrm{WL}})_{\mathcal{D}_{\mathrm{te}},\mathcal{L}} = 0.1\mathbf{H}_{\mathrm{shape},\mathcal{D}_{\mathrm{te}},\mathcal{L}}.$$

The 240 queries produce only two distinct rows of $(\mathbf{K}_{\mathrm{WL}})_{\mathcal{D}_{\mathrm{te}},\mathcal{L}}$, and class balance within each shape fixes accuracy at 50% for any classifier using only these rows: unique full descriptions yield identical predictor inputs. The selected depth-two kernel retains nonzero background similarities between different sets and thus produces richer source–query inputs.

Table H.2: Variable-degree recovery: Micro-F1 (%) over five graph seeds, reported as mean ± population standard deviation. The depth-five reference has distinct WL descriptions but zero background similarity between instances. WLKS uses the selected depth.

Relation	Label rate	WLKS	WL, depth 5	LGC	RAISE
Signal	2%	48.92 ±1.55	50.00 ±0.00	86.08 ±6.58	97.33 ±0.20
Signal	5%	46.58 ±2.05	50.00 ±0.00	95.17 ±1.20	99.00 ±0.68
Null	2%	50.67 ±2.00	50.00 ±0.00	48.67 ±1.80	49.33 ±1.96
Null	5%	51.42 ±2.02	50.00 ±0.00	48.42 ±2.15	48.17 ±3.16

The training/validation/test split is 720/240/240, with 14/36 observed labels at 2/5% and 72/144 at 10/20% in the regular extension. RAISE retains $\alpha = 0.1$, $\lambda = 0.5$, $q = 0.3$, and one round of self-training. LGC uses zero-diagonal $\mathbf{MAM}^\top$ with native degree and class-probability normalization. WLKS uses normalized final-depth histograms and background weight 0.9. Regular WLKS uses depth five and $C_{\mathrm{svm}} = 5.12$. Variable-degree WLKS searches depths 1–5 and selects depth two, $C_{\mathrm{svm}} = 2.56$. The depth-five reference selects $C_{\mathrm{svm}} = 0.08$. The SVC grid is $\{2^3/100, \ldots, 2^{20}/100\}$, and LGC selects $\mu = 1$ from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100\}$ in each family. Each predictor's global setting maximizes mean validation score over the family's graph–budget conditions.

I Incomplete Observations

At omission level p, each designated set loses $\mathrm{round}(p|S_i|)$ randomly chosen members, keeping at least one; masks are drawn independently of labels, one per labeled-subset seed. Rounding in small sets makes the realized member-weighted fraction differ from p, for example 11.91% for PPI-BP at the 10% level. Table I.1 reports the four-task means. Across the five levels, RAISE's four-task micro-F1 lead over PADEL is positive for every labeled-subset mean, and at 20% omission RAISE exceeds PADEL by 5.36 micro-F1 points and 7.58 macro-F1 points.

Table I.1: Member omission on four tasks at 5% labels with the original five labeled subsets. Each method rebuilds its inputs and reruns its fitting procedure on the same observed memberships. Entries are mean Micro-F1 (%). S2N and WLKS are evaluated at 0/10/20% omission.

	Members omitted				
Method	0%	5%	10%	15%	20%
WLKS	41.74	–	40.62	–	39.22
S2N	41.58	–	40.91	–	40.08
PADEL	48.55	48.24	47.38	47.70	47.52
RAISE	54.21	53.76	52.80	53.75	52.88