GRAPH DOMAIN ADAPTATION DOES NOT END WITH REPRESENTATION LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph domain adaptation (GDA) transfers knowledge from a labeled source graph to an unlabeled target graph under shifts in both node attributes and graph structure. Existing methods primarily adapt graph representations through propagation redesign, distribution alignment, or source-to-target transition modeling, but still rely on a single graph-propagating path for target prediction. This leaves open whether an adapted graph representation exhausts the predictive evidence available in the target domain, since the graph-aware expert and graph-free local expert may exhibit different failure modes under topological shifts. To address this limitation, we propose **EviGDA**, an Evidence-Augmented Graph Domain Adaptation framework that complements graph representation adaptation with a graph-free local expert. The graph-aware expert performs message passing and entropy-aware marginal alignment, while the graph-free local expert learns solely from source node features and labels without graph propagation or target alignment. The two experts are optimized independently and combined only at inference through a task-level constant probability mixture, preserving complementary evidence without joint training, learned routing, or target pseudo-labels. Extensive experiments on ten datasets and 16 transfer tasks show that EviGDA outperforms state-of-the-art baselines. Complementary prediction paths add value beyond graph representation alignment.

1 INTRODUCTION

Graph-structured data often exhibit distribution shifts across domains, including changes in node attributes, class proportions, and neighborhood relations. Models trained on a labeled source graph can therefore generalize poorly to an unlabeled target graph. Graph domain adaptation (GDA) addresses this problem by transferring knowledge across graphs with different attribute and structural distributions (Wu et al., 2020; 2023; Liu et al., 2023; 2024b). A predominant line of work approaches GDA through representation adaptation. Source and target embeddings are aligned by adversarial objectives or explicit discrepancy measures (Wu et al., 2020; Dai et al., 2023; Wu et al., 2023; Gretton et al., 2012); subsequent methods account for conditional structure and label shifts (Liu et al., 2023; 2024b), propagation and target smoothness (Liu et al., 2024a; Chen et al., 2025), or attribute, spectral, and homophily discrepancies (You et al., 2023; Fang et al., 2025a;b; Yang et al., 2025). Recent approaches learn adaptive alignment criteria or source-to-target evolution (Chen et al., 2026a;b).

Despite these advances, most GDA methods ultimately infer target labels from a **single graph-propagating prediction path**. Their improvements act on the representation delivered to this path, leaving the final prediction restricted to the information encoded after neighborhood aggregation. This restriction may be consequential under structural shift: aggregation couples node attributes to target-domain connectivity, whereas some attribute–label relations may remain useful without that connectivity (Liu et al., 2023; 2024b; Fang et al., 2025b; Tai et al., 2026). Representation adaptation and predictive sufficiency are therefore different questions. Even an improved graph representation need not exhaust the evidence available for target prediction, while the role of residual evidence outside this representation remains underexplored in GDA.

To address this limitation, we propose **EviGDA**, an Evidence-Augmented Graph Domain Adaptation framework that extends GDA beyond representation adaptation. A graph-aware expert performs domain adaptation using node attributes and connectivity, while a graph-free local expert learns from labeled source attributes alone. The two experts have disjoint parameters, objectives, and

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.

optimizers, and their predictions are combined only after training through a task-level probability mixture. The design keeps their information access distinct and permits a direct test of whether local-expert predictions complement the adapted graph expert. Within the graph-aware expert, we further introduce entropy-aware exact-sampling alignment. It assigns nonzero sampling probability to every target node while favoring lower-entropy predictions, thereby changing target participation without changing the underlying multi-kernel discrepancy estimator (Gretton et al., 2012). This change of empirical measure separates node importance from discrepancy design and concentrates repeated alignment estimates on more reliable target evidence.

To characterize the value of the local expert, we provide two complementary results. First, the gap in minimum achievable log-risk between conditioning on the adapted graph state and additionally conditioning on node attributes equals their residual conditional information (Goldfeld & Polyanskiy, 2020). Second, for a trained expert pair, an exact Brier-risk decomposition shows that, when the graph-free local expert is weaker, some nonzero convex mixture improves upon the graph-aware expert if and only if their expected squared probabilistic disagreement exceeds their Brier-risk gap. The first result identifies the information available beyond the adapted state; the second determines whether the trained experts realize enough of that difference to improve the mixture. Together, they formalize our central insight: graph domain adaptation does not end with representation learning.

We evaluate EviGDA in the standard 16-transfer setting across four benchmark families used by recent GDA studies (Liu et al., 2024a; Chen et al., 2026a). EviGDA establishes a new state of the art by outperforming existing baselines. Prediction-level correctness decompositions reveal complementary errors between the two experts. Under controlled topology corruption, degradation of the graph-aware expert is accompanied by an increase in local-exclusive correctness. Capacity- and ensemble-matched controls distinguish these gains from extra parameters and same-expert averaging.

Our contributions are threefold:

- We propose EviGDA, an independent dual-expert framework that complements graph representation adaptation with a graph-free local expert to exploit evidence beyond the adapted representation.
- We theoretically characterize when graph-free local evidence improves an adapted graph expert and introduce entropy-aware exact-sampling alignment to strengthen graph adaptation.
- Extensive experiments on ten datasets and 16 transfer tasks show that EviGDA outperforms state-of-the-art baselines and consistently benefits from complementary graph-free local evidence.

2 Related Work

Representation-centric graph domain adaptation. Unsupervised domain adaptation commonly reduces the discrepancy between labeled source and unlabeled target representations through adversarial learning, moment matching, or kernel mean embedding (Ben-David et al., 2010; Ganin et al., 2016; Gretton et al., 2012; Long et al., 2015; Sun & Saenko, 2016; Tzeng et al., 2017). Graph domain adaptation further needs to account for relational dependence and structural shifts. Early cross-network approaches such as DANE and ACDNE combine transferable node encoders with adversarial distribution alignment (Zhang et al., 2019; Shen et al., 2020). UDA-GCN and AdaGCN incorporate adversarial adaptation into graph encoders, while GRADE characterizes non-IID graph transfer through representation-level discrepancies (Wu et al., 2020; Dai et al., 2023; Wu et al., 2023). Subsequent methods make representation transfer sensitive to changes in graph structure: StruRW reweights source neighborhoods under conditional structure shift, Pair-Align jointly addresses conditional structure and label shifts, A2GNN adapts propagation depth across source and target graphs, TDSS improves target smoothness through sampled neighborhoods, and SpecReg derives transfer-oriented spectral regularizers (Liu et al., 2023; 2024b;a; Chen et al., 2025; You et al., 2023). Together, these methods progressively strengthen how transferable graph representations are learned. The prediction endpoint itself has received less attention: target decisions are generally derived from the resulting graph-propagating representation. Our work examines complementary predictive evidence retained by a separate graph-free local expert after graph representation adaptation.

Decoupling graph shifts and adaptive transfer. Recent GDA methods further distinguish different sources or dynamics of domain shift. Generative GDA disentangles semantic, domain, and nuisance factors (Cai et al., 2024); GraphAlign instead modifies and compresses the source graph according to

alignment and rescaling principles (Huang et al., 2024); SA-GDA performs category-aware spectral augmentation and combines local and global graph views (Pang et al., 2023); GAA models attribute-driven transfer through interacting attribute and structural channels (Fang et al., 2025a); HGDA explicitly aligns homophily with mixed graph filters (Fang et al., 2025b); JDA-GCN augments adversarial alignment with structural consistency (Yang et al., 2024); and DGSDA separates attribute and topology adaptation through learnable spectral filters (Yang et al., 2025). ADAlign learns an adaptive characteristic-function discrepancy, whereas DiffGDA models continuous structure–semantic evolution from source to target (Chen et al., 2026a;b). Most closely to our concern with neighborhood dependence, DFT decorrelates features inside graph layers to reduce the conditional shift induced by local dependencies (Tai et al., 2026). These methods ask which graph factors should be transferred and how they should be aligned. We address a complementary question: whether the adapted graph path should remain the sole prediction endpoint. Accordingly, our graph-free local expert does not modify the graph encoder or construct an auxiliary graph; it preserves a separate source-supervised view that never accesses adjacency. Within the graph path, ADAlign adapts the discrepancy in spectral-frequency space, whereas our EAM keeps multi-kernel MMD fixed and adapts the empirical target measure through prediction-dependent exact sampling. It changes node importance in alignment rather than introducing another representation view or discrepancy.

Multiple prediction paths and expert complementarity. Mixture-of-experts models learn dense or sparse input-dependent routing among specialized predictors, while deep ensembles aggregate independently trained models to improve predictive robustness (Jacobs et al., 1991; Shazeer et al., 2017; Lakshminarayanan et al., 2017). Under heterophily, H2GCN separates ego and neighbor embeddings to preserve their distinct information (Zhu et al., 2020). In graph learning, Mowst combines an MLP and a GNN through confidence-based node-wise cooperation, and GraphBridge augments a transferred GNN with a trainable side network (Zeng et al., 2024; Ju et al., 2025). EviGDA instead studies post-adaptation evidence under unsupervised GDA. It preserves two independently optimized experts with different information access: one uses attributes and connectivity, while the other never performs graph propagation. Unlike Mowst, their task-level mixture is not produced by a node-wise router; unlike GraphBridge, the graph-free local expert is not trained through a transfer, side-tuning, or fusion objective. The experts interact only after training. Parameter-matched graph models and independently initialized graph–graph ensembles then distinguish heterogeneous predictive complementarity from capacity and generic ensembling.

3 METHOD

Let $\mathcal{G}_s = (\mathbf{X}_s, \mathbf{A}_s, \mathbf{y}_s)$ and $\mathcal{G}_t = (\mathbf{X}_t, \mathbf{A}_t)$ denote the labeled source and unlabeled target graphs, where $\mathbf{X}_d \in \mathbb{R}^{N_d \times D}$ and $\mathbf{A}_d \in \mathbb{R}_{\geq 0}^{N_d \times N_d}$ are the node attributes and processed adjacency matrix of domain $d \in \{s, t\}$. The two domains share K classes, and only the source labels $\mathbf{y}_s \in \{1, \ldots, K\}^{N_s}$ are available during training. Figure 1 summarizes the graph-aware expert, independently trained graph-free local expert, and post-training probability fusion in EviGDA.

3.1 GRAPH-AWARE AND GRAPH-FREE LOCAL EXPERTS

We first obtain graph-aware node predictions using A2GNN (Liu et al., 2024a). Its encoder applies P_d propagation steps in domain d, allowing different source and target depths. Propagation uses symmetric degree normalization with missing self-loops added. One feature transformation and ReLU produce the embeddings, followed by one graph classification layer; $P_d = 0$ therefore removes encoder propagation but retains the graph classifier:

$$\mathbf{H}_d^g = F_g(\mathbf{X}_d, \mathbf{A}_d; P_d), \qquad \mathbf{\Pi}_d^g = C_g(\mathbf{H}_d^g, \mathbf{A}_d), \qquad \mathbf{\Pi}_d^\ell = F_\ell(\mathbf{X}_d). \tag{1}$$

Here $\mathbf{H}_d^g \in \mathbb{R}^{N_d \times h_g}$ contains the graph-aware node embeddings, $\mathbf{h}_{d,i}^g$ denotes its ith row, and $\mathbf{\Pi}_d^g, \mathbf{\Pi}_d^\ell \in \mathbb{R}^{N_d \times K}$ contain class probabilities. The experts share source labels but deliberately differ in information access: $F_g : (\mathbf{X}_d, \mathbf{A}_d) \mapsto \mathbf{H}_d^g$, whereas $F_\ell : \mathbf{X}_d \mapsto \mathbf{\Pi}_d^\ell$ and therefore $\partial \mathbf{\Pi}_d^\ell / \partial \mathbf{A}_d = \mathbf{0}$. Message passing composes attributes with neighborhood evidence, which is useful when connectivity transfers but can obscure stable attribute cues under structural shift. The graph-free local expert is implemented as a feature-only two-layer MLP that never reads $\mathbf{A}_d$.

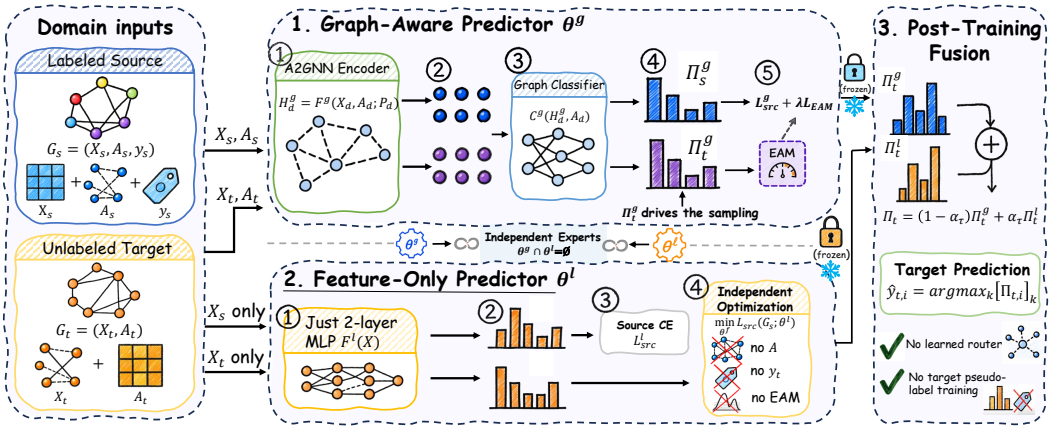


Figure 1: **Overview of EviGDA.** The graph-aware expert is optimized by source classification and entropy-aware exact-sampling alignment. The graph-free local expert receives source supervision alone. Their parameters remain independent, and their target probabilities are combined after training using one coefficient shared by all nodes in the transfer task.

The two experts also have disjoint parameters, $\boldsymbol{\theta}_g \cap \boldsymbol{\theta}_\ell = \varnothing$, with $\nabla_{\boldsymbol{\theta}_\ell} \mathbf{\Pi}_d^g = \mathbf{0}$ and $\nabla_{\boldsymbol{\theta}_g} \mathbf{\Pi}_d^\ell = \mathbf{0}$. We denote their source cross-entropies by $\mathcal{L}_{\text{src}}^g$ and $\mathcal{L}_{\text{src}}^\ell$; the expert-specific objectives are given after the alignment mechanism is defined. Their predictions interact only after training, in probability space. A second head on $\mathbf{H}_d^g$ would inherit the same adapted state and primarily test capacity or generic ensembling. In contrast, F_ℓ preserves a separate attribute-based path; Section 4.6 tests this distinction with controls for parameter capacity and the choice of prediction path.

3.2 Why Representation Learning Need Not Be Sufficient

Let $(X, Y, Z^g) \sim P_t$ describe a random target node, where X and Y are its attributes and label and Z^g is the state available to the graph classifier after message passing. Target sufficiency requires $P_t(Y \mid Z^g, X) = P_t(Y \mid Z^g)$ almost surely. For any input S, define $\mathcal{R}_{\log}^\star(S) = \inf_q \mathbb{E}_t[-\log q(Y \mid S)]$. For predictions, let $\mathcal{R}(\boldsymbol{\pi}) = \mathbb{E}_t\|\mathbf{e}_Y - \boldsymbol{\pi}\|_2^2$ be the Brier risk (Gneiting & Raftery, 2007), for one-hot labels $\mathbf{e}_Y$. Set $\mathcal{R}_b = \mathcal{R}(\boldsymbol{\pi}^b)$ for $b \in \{g, \ell\}$, $\Delta = \mathcal{R}_\ell - \mathcal{R}_g$, and $D = \mathbb{E}_t\|\boldsymbol{\pi}^g - \boldsymbol{\pi}^\ell\|_2^2$.

Proposition 1 (Residual information and weak-expert gain)**.** *The representation-level risk gap and the realized mixture risk satisfy*

$$\mathcal{R}_{\log}^\star(Z^g) - \mathcal{R}_{\log}^\star(Z^g, X) = I_t(Y; X \mid Z^g) \geq 0, \tag{2}$$

$$\mathcal{R}(\boldsymbol{\pi}^\alpha) - \mathcal{R}_g = \alpha(\Delta - D) + \alpha^2 D, \quad \boldsymbol{\pi}^\alpha = (1-\alpha)\boldsymbol{\pi}^g + \alpha\boldsymbol{\pi}^\ell. \tag{3}$$

The first equality is zero if and only if $Y \perp X \mid Z^g$. If $\Delta \geq 0$, some $\alpha \in (0, 1]$ strictly improves the graph-aware expert if and only if $D > \Delta$; then

$$\alpha^\star = \frac{D - \Delta}{2D}, \qquad \mathcal{R}_g - \mathcal{R}(\boldsymbol{\pi}^{\alpha^\star}) = \frac{(D-\Delta)^2}{4D} > 0. \tag{4}$$

The first identity follows because Bayes log-risk equals conditional entropy (Goldfeld & Polyanskiy, 2020); the second specializes the ensemble ambiguity decomposition (Krogh & Vedelsby, 1994) to two experts. Conditional information identifies evidence beyond Z^g, while $D > \Delta$ tests whether the local expert can improve a fixed mixture. If $D \leq \Delta$, $\alpha = 0$ is optimal. We evaluate Macro-F1 empirically and provide the complete proper-risk proofs in Appendix A.1.

3.3 Entropy-Aware Exact-Sampling Alignment

Conventional marginal alignment samples target nodes uniformly and therefore treats them as equally informative. Under domain shift, uncertain graph predictions may reflect unstable neighborhood evidence and affect adaptation as often as confident ones. EAM retains the kernel discrepancy but reallocates sampling mass toward lower-entropy target nodes before measuring the cross-domain

discrepancy. For target node i, let $\boldsymbol{\pi}^g_{t,i}$ be the ith row of $\boldsymbol{\Pi}^g_t$. Its normalized predictive entropy and sampling probability are defined as follows. The stop-gradient operator $\mathrm{sg}[\cdot]$ preserves forward values but blocks backpropagation through the sampling probabilities.

$$e_i = -\frac{1}{\log K}\sum_{k=1}^{K}\pi^g_{t,ik}\log\pi^g_{t,ik}, \qquad q_i = \mathrm{sg}\left[\frac{\varepsilon+(1-\varepsilon)(1-e_i)}{\sum_{j=1}^{N_t}[\varepsilon+(1-\varepsilon)(1-e_j)]}\right]. \tag{5}$$

We use $0\log 0 = 0$ and a floor $\varepsilon \in (0,1]$ to give every target node nonzero sampling probability. For $\varepsilon < 1$, lower-entropy predictions receive more mass; $\varepsilon = 1$ recovers uniform sampling. The two laws define entropy-tilted and uniform empirical target measures:

$$\widehat{\mu}^{\mathrm{EAM}}_t = \sum_{i=1}^{N_t} q_i\,\delta_{\mathbf{h}^g_{t,i}}, \qquad \widehat{\mu}^{\mathrm{unif}}_t = \frac{1}{N_t}\sum_{i=1}^{N_t}\delta_{\mathbf{h}^g_{t,i}}. \tag{6}$$

Here $\delta_{\mathbf{h}}$ denotes a point mass at embedding $\mathbf{h}$. Reliability controls repeated participation, while the positive floor gives every target node a nonzero probability of contributing to alignment.

At repetition r, we draw m source nodes uniformly and m target nodes from $\mathbf{q}$, both with replacement. Let $\mathbf{s}_{r,a} = \mathrm{sg}[\mathbf{h}^g_{s,I^s_{r,a}}]$ and $\mathbf{t}_{r,a} = \mathbf{h}^g_{t,I^t_{r,a}}$ denote the ath sampled source and target embeddings, respectively. With repetition-specific kernel k_r, EAM averages their discrepancies over R draws:

$$\mathcal{L}_{\mathrm{EAM}} = \frac{1}{Rm^2}\sum_{r=1}^{R}\sum_{a,b=1}^{m}\left[k_r(\mathbf{s}_{r,a},\mathbf{s}_{r,b}) + k_r(\mathbf{t}_{r,a},\mathbf{t}_{r,b}) - 2k_r(\mathbf{s}_{r,a},\mathbf{t}_{r,b})\right]. \tag{7}$$

The source–source and target–target sums retain pairs with $a = b$, matching the biased kernel estimator used in our implementation (Gretton et al., 2012). EAM changes the target samples entering this estimator while leaving its kernel form unchanged; categorical resampling is a Monte Carlo realization of $\widehat{\mu}^{\mathrm{EAM}}_t$, not a continuous weighted-MMD surrogate. Source embeddings are detached, whereas target embeddings retain gradients. For each repetition, k_r sums five RBF kernels with bandwidths $b_r 2^{j-2}$, $j = 0,\ldots,4$. Here b_r is the detached mean squared distance between distinct pooled sample positions, clamped away from zero. Both variants use $m = \min(1000, N_s, N_t)$ samples and $R = 5$ repetitions in every transfer task.

3.4 Independent Training and Post-training Fusion

Let $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_\ell$ denote the disjoint parameter sets of the graph-aware and graph-free local experts. For $b \in \{g, \ell\}$, source supervision is measured by

$$\mathcal{L}^b_{\mathrm{src}} = -\frac{1}{N_s}\sum_{i=1}^{N_s}\log[\boldsymbol{\pi}^b_{s,i}]_{y_{s,i}}. \tag{8}$$

For $\lambda \geq 0$, the two parameter sets are optimized with

$$\min_{\boldsymbol{\theta}_g}\left[\mathcal{L}^g_{\mathrm{src}}(\mathcal{G}_s;\boldsymbol{\theta}_g) + \lambda\mathcal{L}_{\mathrm{EAM}}(\mathcal{G}_s,\mathcal{G}_t;\boldsymbol{\theta}_g)\right], \qquad \min_{\boldsymbol{\theta}_\ell}\mathcal{L}^\ell_{\mathrm{src}}(\mathcal{G}_s;\boldsymbol{\theta}_\ell). \tag{9}$$

Here λ balances source classification and graph-expert alignment; the graph-free local expert receives source supervision alone. The graph objective may use both $\mathcal{G}_s$ and $\mathcal{G}_t$, but target labels occur in neither objective. Under the gradient convention in Section 3.3, EAM reaches $\boldsymbol{\theta}_g$ through sampled target embeddings while its source embeddings are fixed anchors. No term sends an EAM gradient to $\boldsymbol{\theta}_\ell$. Thus, independent training means disjoint parameters, optimizers, and objectives—not merely two classifier heads attached to one adapted representation.

For transfer $\tau = (s \to t)$, let $\boldsymbol{\pi}^g_{\tau,i}$ and $\boldsymbol{\pi}^\ell_{\tau,i}$ denote the probabilities returned for target node i by the fitted experts. After training, a task-level coefficient $\alpha_\tau \in [0,1]$ forms

$$\boldsymbol{\pi}_{\tau,i} = (1-\alpha_\tau)\boldsymbol{\pi}^g_{\tau,i} + \alpha_\tau\boldsymbol{\pi}^\ell_{\tau,i}, \qquad \widehat{y}_{\tau,i} = \arg\max_k[\boldsymbol{\pi}_{\tau,i}]_k. \tag{10}$$

Since both expert outputs lie in the probability simplex, Eq. 10 is also a valid probability vector for every $\alpha_\tau \in [0,1]$. The same coefficient is shared across target nodes, so fusion introduces neither a node-wise router nor a pseudo-label objective. Fusion is performed after expert training and introduces no additional gradient update. We select the task coefficient from a shared finite grid under the evaluation protocol in Section 4.1 for all 16 transfer directions.

4 EXPERIMENTS

We evaluate EviGDA on 16 transfers to assess adaptation performance and identify the contributions of its two prediction paths and alignment mechanism. The experiments connect expert-level evidence to matched controls, topology interventions, and fusion sensitivity through five research questions:

- **RQ1.** *How effective is EviGDA across graph domains?* Section 4.2 compares its adaptation performance with representative GDA methods on 16 transfers across four benchmark families.
- **RQ2.** *Do the experts provide complementary evidence?* Sections 4.3 and 4.5 analyze expert-specific correctness and controlled topology corruption to examine the value of graph-free predictions.
- **RQ3.** *Does entropy-aware sampling improve alignment?* Section 4.4 compares EAM with matched uniform sampling to isolate the contribution of entropy-guided target participation.
- **RQ4.** *Can capacity or generic ensembling explain the gains?* Section 4.6 compares the heterogeneous pair with parameter-matched Graph-only and Graph+Graph controls.
- **RQ5.** *How sensitive and costly is task-level fusion?* Section 4.6 examines responses to the fusion coefficient and measures the local expert's training, inference, and memory overhead.

4.1 EXPERIMENTAL SETUP

Datasets and metrics. We evaluate ten processed graphs and 16 directed transfer tasks adopted by recent graph domain adaptation studies (Liu et al., 2024a; Chen et al., 2026a). The Citation family contains ACMv9 (A), Citationv1 (C), and DBLPv7 (D); the Airport family contains Brazil (B), Europe (E), and USA (U); the Blog family contains Blog1 (B1) and Blog2 (B2); and the Twitch family contains German (DE) and English (EN). Citation and Airport contribute six directed transfers each, whereas Blog and Twitch contribute two transfers each. All methods read the same processed tensors, preserving the supplied node features and edge representation for each transfer direction.

Macro-F1 is the primary evaluation metric. We additionally report Micro-F1 at the checkpoint selected according to Macro-F1, denoted by Micro@Macro. Family-level scores are equal-task averages within each graph family, and the Full-16 score assigns equal weight to each transfer task. Neither aggregate is weighted by the number of nodes in a graph.

Baselines. We compare against source-only GCN (Kipf & Welling, 2017), alignment-based UDA-GCN (Wu et al., 2020) and GRADE (Wu et al., 2023); graph-shift-aware PairAlign (Liu et al., 2024b), GraphAlign (Huang et al., 2024), A2GNN (Liu et al., 2024a), TDSS (Chen et al., 2025), DGSDA (Yang et al., 2025), GAA (Fang et al., 2025a), and HGDA (Fang et al., 2025b); and recent adaptation methods ADAlign (Chen et al., 2026a), DiffGDA (Chen et al., 2026b), and DFT (Tai et al., 2026). ADAlign and DiffGDA represent recent state-of-the-art approaches to our knowledge. Graph-only serves as our backbone control for measuring gains from the full method.

Implementation and evaluation protocol. Following recent GDA evaluation practice (Chen et al., 2026a), all locally reproduced methods are trained for 150 epochs and evaluated over the same five runs using a unified evaluator; we report the mean and sample standard deviation across runs. EviGDA uses Adam, graph/local hidden widths of 128/64, one graph feature layer, ReLU, and no dropout. Only the graph learning rate, propagation pair, and alignment weight vary during training; all other architecture, regularization, and estimator settings remain fixed. The graph-aware and graph-free local experts are optimized independently, and their output probabilities are combined after training through a task-level constant mixture. Each task evaluates 96 family-level training bundles and replays fusion over a shared coefficient grid. Appendix B.4 lists the family-level search spaces and globally fixed implementation settings. The server has dual Intel Xeon E5-2680 v4 CPUs and 48 GB NVIDIA GeForce RTX 4090 D GPUs; each training run uses one GPU.

Statistical analysis. Controlled contrasts pair task and run, reporting task-level win/tie/loss counts. Expert and capacity comparisons use 10,000 bootstrap replicates, resampling tasks and then paired runs within each task. The 95% intervals span the 2.5th–97.5th percentiles of the aggregate paired-difference distribution, retaining equal task weights in each replicate.

4.2 Overall Adaptation Performance

Adaptation versus source-only learning. First, UDA-GCN and GRADE outperform source-only GCN on all 16 transfers in Table 1. On B1→B2, Macro-F1 increases from 21.96 with GCN to 30.94 and 40.56, respectively. These consistent gains highlight the importance of addressing cross-domain discrepancies: learning a graph classifier from source supervision alone does not ensure that its representations remain predictive in the target domain.

Table 1: **Macro-F1 on 16 graph-domain transfers.** Mean (%) and sample standard deviation over five runs; best and second-best results are bold and underlined.

Method	A→C	A→D	C→A	C→D	D→A	D→C	B1→B2	B2→B1
Source-only								
GCN (ICLR'17)	66.97 ± 1.62	61.38 ± 1.10	66.82 ± 0.85	66.97 ± 0.70	57.34 ± 1.32	65.60 ± 2.53	21.96 ± 1.88	23.05 ± 2.04
Graph domain adaptation								
UDAGCN (WWW'20)	77.62 ± 1.35	71.59 ± 2.31	73.56 ± 1.56	75.17 ± 2.65	62.17 ± 1.98	72.23 ± 2.30	30.94 ± 1.13	31.34 ± 1.15
GRADE (AAAI'23)	68.06 ± 2.29	63.94 ± 2.22	68.46 ± 0.40	69.77 ± 1.17	63.76 ± 0.61	65.74 ± 0.95	40.56 ± 1.37	42.41 ± 2.76
PairAlign (ICML'24)	57.09 ± 1.61	55.25 ± 2.05	52.20 ± 2.50	56.73 ± 3.25	49.34 ± 1.36	52.31 ± 2.27	40.99 ± 2.07	43.73 ± 0.68
GraphAlign (KDD'24)	67.07 ± 1.29	63.05 ± 1.79	63.57 ± 0.90	65.99 ± 1.46	59.02 ± 0.97	62.15 ± 0.93	35.85 ± 1.85	39.60 ± 1.23
A2GNN-MMD (AAAI'24)	78.96 ± 0.42	72.66 ± 0.46	75.17 ± 0.28	74.70 ± 0.73	73.96 ± 0.61	78.11 ± 0.98	44.15 ± 2.30	43.77 ± 0.83
TDSS-RW (AAAI'25)	79.67 ± 0.19	73.71 ± 0.69	76.16 ± 0.42	72.72 ± 0.42	74.70 ± 0.46	76.43 ± 0.27	42.81 ± 1.15	45.11 ± 1.28
GAA (ICLR'25)	71.97 ± 0.66	64.84 ± 1.36	69.32 ± 0.35	69.88 ± 0.88	63.00 ± 0.57	68.73 ± 1.68	44.61 ± 2.34	40.79 ± 1.39
DGSDA (ICML'25)	80.54 ± 0.39	74.13 ± 1.01	76.04 ± 0.19	75.89 ± 0.22	74.90 ± 0.36	**81.49** ± 0.50	36.33 ± 1.70	39.67 ± 2.38
HGDA (ICML'25)	71.58 ± 0.96	65.00 ± 1.03	67.59 ± 0.77	69.09 ± 0.97	66.21 ± 0.83	70.86 ± 0.99	43.32 ± 0.40	43.64 ± 0.36
ADAlign (ICLR'26)	79.03 ± 0.38	75.14 ± 1.55	75.25 ± 0.37	76.59 ± 1.01	72.15 ± 1.31	75.40 ± 1.70	40.81 ± 2.52	40.68 ± 1.09
DiffGDA (ICLR'26)	78.83 ± 1.09	72.67 ± 0.89	74.88 ± 0.86	73.33 ± 1.21	71.26 ± 2.70	75.45 ± 1.44	42.69 ± 2.07	40.15 ± 2.88
DFT (KDD'26)	72.59 ± 1.83	69.69 ± 1.02	68.06 ± 0.73	70.90 ± 1.92	65.31 ± 2.71	75.35 ± 1.92	47.25 ± 8.35	51.75 ± 4.24
EviGDA (Ours)	**81.93** ± 0.28	**76.18** ± 1.48	**77.34** ± 0.19	**78.08** ± 0.41	**75.48** ± 0.15	81.24 ± 0.54	**50.88** ± 1.73	**53.01** ± 0.61
Method	U→B	U→E	B→U	B→E	E→U	E→B	DE→EN	EN→DE
Source-only								
GCN (ICLR'17)	45.06 ± 3.21	34.17 ± 2.12	29.41 ± 0.32	25.27 ± 0.86	30.30 ± 0.22	32.66 ± 1.84	56.23 ± 0.32	57.31 ± 1.86
Graph domain adaptation								
UDAGCN (WWW'20)	60.82 ± 22.45	41.28 ± 1.77	39.41 ± 2.55	50.03 ± 0.80	40.35 ± 1.06	63.89 ± 1.69	57.91 ± 0.47	58.33 ± 1.54
GRADE (AAAI'23)	61.48 ± 4.34	47.61 ± 1.00	38.50 ± 3.07	55.45 ± 1.90	43.96 ± 1.39	70.38 ± 1.48	58.72 ± 0.06	62.12 ± 0.17
PairAlign (ICML'24)	70.03 ± 1.82	39.29 ± 1.42	45.35 ± 4.19	38.88 ± 3.89	40.50 ± 2.84	48.45 ± 3.02	60.11 ± 0.37	63.72 ± 0.16
GraphAlign (KDD'24)	63.69 ± 2.27	53.24 ± 1.20	46.21 ± 2.86	54.18 ± 1.59	48.83 ± 2.61	68.54 ± 1.02	54.27 ± 1.01	55.07 ± 2.61
A2GNN-MMD (AAAI'24)	60.36 ± 1.56	46.52 ± 1.03	43.10 ± 1.81	48.12 ± 1.06	40.28 ± 5.22	64.11 ± 6.61	56.71 ± 0.48	56.15 ± 1.49
TDSS-RW (AAAI'25)	74.56 ± 0.60	41.89 ± 0.28	53.96 ± 1.68	43.26 ± 3.02	48.23 ± 0.07	59.29 ± 0.36	53.62 ± 0.91	40.07 ± 1.67
GAA (ICLR'25)	70.42 ± 1.34	54.02 ± 3.25	55.66 ± 0.73	53.40 ± 2.51	50.68 ± 1.20	71.48 ± 1.88	46.11 ± 1.75	46.45 ± 1.02
DGSDA (ICML'25)	63.37 ± 1.30	52.87 ± 1.20	56.17 ± 0.35	51.87 ± 1.32	50.75 ± 0.39	70.36 ± 1.12	59.92 ± 0.12	61.49 ± 0.18
HGDA (ICML'25)	71.83 ± 22.93	43.56 ± 3.34	52.91 ± 0.95	50.49 ± 2.00	48.12 ± 0.71	66.11 ± 1.82	59.52 ± 0.34	60.84 ± 0.64
ADAlign (ICLR'26)	75.56 ± 2.31	52.23 ± 1.29	47.86 ± 1.36	55.13 ± 1.25	50.27 ± 0.79	67.87 ± 1.27	59.33 ± 0.10	62.94 ± 0.40
DiffGDA (ICLR'26)	75.73 ± 1.41	47.07 ± 1.34	45.64 ± 0.94	53.09 ± 0.82	41.07 ± 1.38	62.70 ± 0.97	56.02 ± 0.23	57.65 ± 1.34
DFT (KDD'26)	76.62 ± 0.46	50.11 ± 1.15	53.58 ± 1.16	53.32 ± 1.78	49.52 ± 1.09	69.10 ± 1.04	60.55 ± 0.60	63.63 ± 1.21
EviGDA (Ours)	**81.20** ± 0.35	**58.76** ± 0.60	**56.90** ± 0.32	**59.76** ± 0.14	**51.41** ± 1.08	**75.75** ± 1.30	**60.58** ± 0.09	**64.22** ± 0.41

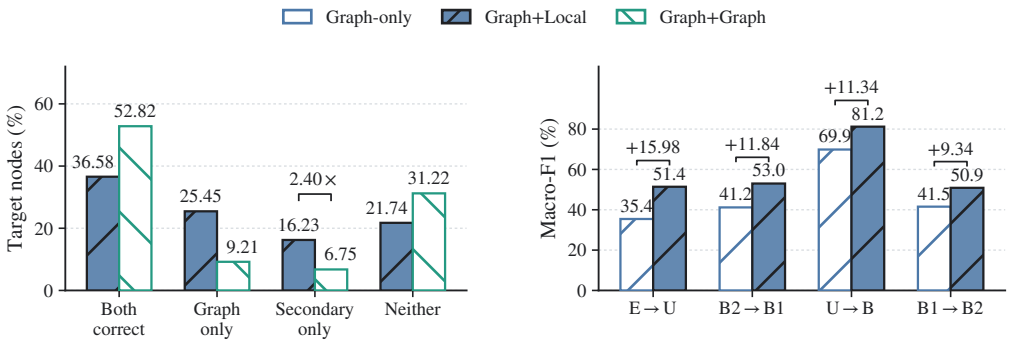
Progress in graph adaptation. Second, recent methods further improve representative transfer directions. Compared with UDA-GCN, ADAlign's adaptive alignment raises C→D from 75.17 to 76.59, while DGSDA's separation of attribute and topology adaptation raises D→C from 72.23 to 81.49. These improvements suggest that effective transfer depends not only on reducing distribution discrepancy, but also on how attributes and relational information are used. This motivates examining whether complementary information should remain separately accessible at prediction time, rather than deriving every target decision from one adapted graph representation.

Complementary prediction paths. Finally, EviGDA achieves 67.67 ± 0.04 Full-16 Macro-F1 and ranks first on 15 transfers, including all six Airport and both Blog and Twitch directions. It exceeds the strongest competing results by 4.74 points on U→E and 4.27 on E→B. This success is attributed to preserving two complementary prediction paths: the graph-aware expert captures relational information, while the graph-free local expert retains attribute evidence without neighborhood aggregation. Constant probability fusion combines these predictions, and entropy-aware sampling prioritizes confident target evidence during graph alignment. The following controls examine the contributions of expert complementarity and sampling importance separately.

4.3 Complementary Predictive Evidence

Figure 2 partitions post-training target correctness for Local or Graph 2 against the same graph-aware expert and compares Graph-only with fused endpoints on representative transfers.

Fusion exceeds the better single expert by 1.98 points (95% CI $[0.67, 3.67]$), with all five run-level aggregates positive. The local expert is exclusively correct on 16.23% of target nodes, more than



(a) Exclusive predictive evidence **(b) Benefit of heterogeneous fusion**

Figure 2: **Heterogeneous predictive complementarity.** (a) Full-16 correctness: Local supplies 2.40× the exclusive correct predictions of Graph 2 against the same graph expert. (b) Representative transfers: Graph-only versus Graph+Local, with bracketed fusion gains in Macro-F1 points.

twice the 6.75% supplied by a second graph expert. Fusion corrects graph errors on 10.87% of target nodes and introduces errors on 4.45%. The resulting gain reflects useful local-exclusive predictions on nodes where the graph-aware expert is incorrect.

Graph+Graph instead yields -0.82% net repairs, indicating that independent initialization alone need not provide useful corrections to the graph expert's errors.

The corresponding fusion gains are 0.05, 8.21, 10.59, and 1.42 points on Citation, Airport, Blog, and Twitch. The larger Airport and Blog responses align with greater local-exclusive evidence. We examine this relationship directly through the degree-preserving topology intervention in Section 4.5.

4.4 Entropy-Aware Alignment

We test whether predictive confidence informs target participation in marginal alignment. EAM changes the empirical target measure, not the discrepancy objective: lower-entropy predictions recur more often across draws, while the sampling floor retains every node's support. Thus, node participation changes while the discrepancy criterion remains fixed.

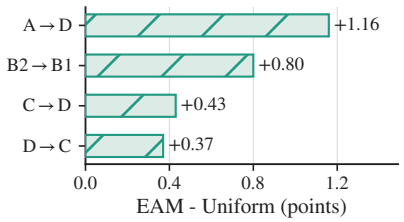


Figure 3: **Matched entropy-aware resampling.** EAM gains on representative transfers; Appendix D.3 reports all 16 matched task comparisons.

Matched comparison. The Uniform variant assigns every target node the same sampling probability while leaving the experts, sample count, replacement semantics, multi-kernel MMD, bandwidth rule, optimizer, and source/target gradient routes unchanged throughout training.

Alignment contribution. EAM improves the fused endpoint by 0.24 Macro-F1 points on average, with family effects of $+0.35$, $+0.06$, $+0.59$, and $+0.08$ on Citation, Airport, Blog, and Twitch. Figure 3 shows gains of 1.16, 0.80, 0.43, and 0.37 points on A→D, B2→B1, C→D, and D→C. Entropy-guided participation strengthens these alignment estimates by concentrating repeated draws on confident target evidence from the graph expert.

4.5 Controlled Topology-Shift Stress Test

We progressively rewire target edges on A→C, U→B, B1→B2, and EN→DE, using six severity levels and five runs per transfer. Rewiring preserves edge count and every node's directed degree while leaving attributes and local predictions unchanged. Figure 4 tracks how this controlled neighborhood corruption changes graph-expert performance and local-exclusive evidence.

Homophily loss tracks graph damage ($\rho = 0.943$), which in turn tracks local-exclusive evidence ($\rho = 0.973$). At maximum corruption, Graph-only drops by 0.270 while the local-exclusive rate rises by 0.141. Under this matched intervention, the trajectories isolate the increasing relative value of

topology-free evidence. A→C responds most strongly and U→B least, showing that the value of topology-free evidence depends on the transfer's response to neighborhood corruption.

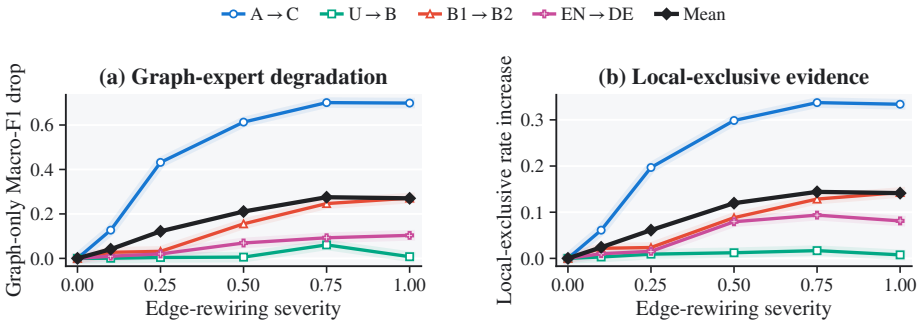


Figure 4: **Topology corruption exposes heterogeneous evidence.** Graph-expert degradation and local-exclusive correctness under degree-preserving edge rewiring; colored curves denote representative transfers, with their equal-task mean shown as the black reference curve.

4.6 Controlled Attribution

We compare with a graph model widened to match Full within five parameters and Graph+Graph, which replaces Local with an independently initialized graph expert. The graph pair preserves ensembling while giving both experts the same inputs. All reviewer controls use the frozen task configuration of the full method, the same training and checkpoint budgets, and identical evaluation procedures. Only the expert identity or graph-model capacity is changed. Graph+Local gains 5.13 points over the capacity control (14 task wins), 4.84 over Graph+Graph (12 wins, four ties), and 1.98 over the better single expert (Table 2). Together with Figure 2(a), these controls support the value of heterogeneous information access beyond added capacity and generic averaging. The local expert retains an attribute-only prediction path alongside the adapted graph representation, allowing task-level probability fusion to exploit complementary predictions from independently trained experts.

Fusion response. Frozen-logit responses peak at an interior coefficient. E→U and B2→B1 retain broad beneficial regions; U→B favors graph-dominant fusion. **Efficiency.** Without routing or extra graph propagation, the local expert adds about 2% training time and 9.38% inference latency, with unchanged peak memory on A→C (RTX 4090 D). These final-model costs exclude the one-time hyperparameter search.

Table 2: **Paired gains over capacity and ensemble controls** (percentage points; 95% CIs).

Control	Δ	95% CI	W/T/L
Capacity-matched graph	+5.13	[1.93, 9.05]	14/0/2
Graph+Graph	+4.84	[1.72, 8.60]	12/4/0
Best expert	+1.98	[0.67, 3.67]	12/4/0

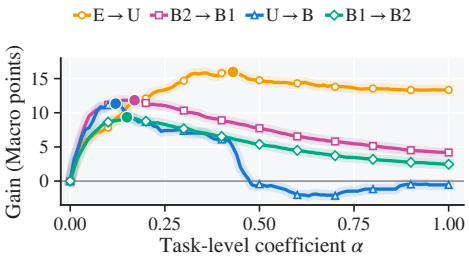


Figure 5: **Task-level fusion response.** Curves replay frozen expert predictions; filled markers denote observed maxima. Soft under-strokes emphasize the response trajectories.

5 Conclusion

EviGDA complements representation alignment with independently trained graph-aware and graph-free local experts, entropy-aware exact sampling, and task-level probability fusion. It achieves the best Macro-F1 on 15 of 16 transfers and improves over the stronger expert by 1.98 points on average. Capacity and ensemble controls support heterogeneous information access as the source of these gains. The current study focuses on static graphs with a single labeled source domain. Future work will support evolving graphs through incremental expert updates and refreshed alignment samples, and multi-source adaptation through source-specific graph experts and source-level probability fusion.

AI USE STATEMENT

Generative AI tools were used to assist with literature organization, research brainstorming, code review, result-consistency checking, scientific-figure preparation, and manuscript editing. The authors designed the methods and experimental protocols, executed all experiments, verified the reported measurements against the underlying artifacts, checked the cited literature, and made all final scientific and editorial decisions. No generative AI tool was used to generate benchmark data, target labels, or experimental measurements. The authors take full responsibility for the contents of this paper.

REPRODUCIBILITY STATEMENT

Section 3 specifies the model architecture, optimization objectives, gradient paths, entropy-aware sampling procedure, and post-training fusion rule. Section 4 describes the datasets, evaluation metrics, baseline comparisons, repeated runs, and statistical analyses. The appendix provides estimator details, preprocessing conventions, additional results, matched-control specifications, parameter analyses, and computational-cost measurements. The supplementary material provides the implementation and scripts used to reproduce the reported tables and figures.

REFERENCES

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1–2):151–175, 2010.

Ruichu Cai, Fengzhu Wu, Zijian Li, Pengfei Wei, Lingling Yi, and Kun Zhang. Graph domain adaptation: A generative view. *ACM Transactions on Knowledge Discovery from Data*, 18(3), 2024.

Wei Chen, Guo Ye, Yakun Wang, Zhao Zhang, Libang Zhang, Daixin Wang, Zhiqiang Zhang, and Fuzhen Zhuang. Smoothness really matters: A simple yet effective approach for unsupervised graph domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 15875–15883, 2025.

Wei Chen, Xingyu Guo, Shuang Li, Zhao Zhang, Yan Zhong, Fuzhen Zhuang, and Deqing Wang. Learning adaptive distribution alignment with neural characteristic function for graph domain adaptation. In *International Conference on Learning Representations*, 2026a.

Wei Chen, Xingyu Guo, Shuang Li, Yan Zhong, Zhao Zhang, Fuzhen Zhuang, Hongrui Liu, Libang Zhang, Guo Ye, and Huimei He. Learning structure-semantic evolution trajectories for graph domain adaptation. In *International Conference on Learning Representations*, 2026b.

Quanyu Dai, Xiao-Ming Wu, Jiaren Xiao, Xiao Shen, and Dan Wang. Graph transfer learning via adversarial domain adaptation with graph convolution. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4908–4922, 2023.

Ruiyi Fang, Bingheng Li, Zhao Kang, Qiuhao Zeng, Nima Hosseini Dashtbayaz, Ruizhi Pu, Boyu Wang, and Charles Ling. On the benefits of attribute-driven graph domain adaptation. In *International Conference on Learning Representations*, 2025a.

Ruiyi Fang, Bingheng Li, Jingyu Zhao, Ruizhi Pu, Qiuhao Zeng, Gezheng Xu, Charles Ling, and Boyu Wang. Homophily enhanced graph domain adaptation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 16006–16028, 2025b.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016.

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

Ziv Goldfeld and Yury Polyanskiy. The information bottleneck problem and its applications in machine learning. *IEEE Journal on Selected Areas in Information Theory*, 1(1):19–38, 2020.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Renhong Huang, Jiarong Xu, Xin Jiang, Ruichuan An, and Yang Yang. Can modifying data address graph domain adaptation? In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1131–1142, 2024.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

Li Ju, Xingyi Yang, Qi Li, and Xinchao Wang. Graphbridge: Towards arbitrary transfer learning in gnns. In *International Conference on Learning Representations*, 2025.

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

Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems*, volume 7, 1994.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Meihan Liu, Zeyu Fang, Zhen Zhang, Ming Gu, Sheng Zhou, Xin Wang, and Jiajun Bu. Rethinking propagation for unsupervised graph domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13963–13971, 2024a.

Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. Structural reweighting improves graph domain adaptation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21778–21793, 2023.

Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32552–32575, 2024b.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 97–105, 2015.

Jinhui Pang, Zixuan Wang, Jiliang Tang, Mingyan Xiao, and Nan Yin. SA-GDA: Spectral augmentation for graph domain adaptation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 309–318, 2023.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Xiao Shen, Quanyu Dai, Fu-lai Chung, Wei Lu, and Kup-Sze Choi. Adversarial deep network embedding for cross-network node classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 2991–2999, 2020.

Baochen Sun and Kate Saenko. Deep CORAL: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision Workshops*, pp. 443–450, 2016.

Xinwei Tai, Dongmian Zou, and Hongfei Wang. Enhancing node-level graph domain adaptation by alleviating local dependency. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1366–1377, 2026.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7167–7176, 2017.

Jun Wu, Jingrui He, and Elizabeth A. Ainsworth. Non-iid transfer learning on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 10342–10350, 2023.

Man Wu, Shirui Pan, Chuan Zhou, Xiaojun Chang, and Xingquan Zhu. Unsupervised domain adaptive graph convolutional networks. In *Proceedings of The Web Conference 2020*, pp. 1457–1467, 2020.

Liang Yang, Xin Chen, Jiaming Zhuo, Di Jin, Chuan Wang, Xiaochun Cao, Zhen Wang, and Yuanfang Guo. Disentangled graph spectral domain adaptation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70632–70648, 2025.

Niya Yang, Ye Wang, Zhizhi Yu, Dongxiao He, Xin Huang, and Di Jin. Joint domain adaptive graph convolutional network. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 2496–2504, 2024.

Yuning You, Tianlong Chen, Zhangyang Wang, and Yang Shen. Graph domain adaptation via theory-grounded spectral regularization. In *International Conference on Learning Representations*, 2023.

Hanqing Zeng, Hanjia Lyu, Diyi Hu, Yinglong Xia, and Jiebo Luo. Mixture of weak and strong experts on graphs. In *International Conference on Learning Representations*, 2024.

Yizhou Zhang, Guojie Song, Lun Du, Shuwen Yang, and Yilun Jin. DANE: Domain adaptive network embedding. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4362–4368, 2019.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7793–7804, 2020.

A Theoretical Results

A.1 Prediction-space complementarity

Proof of Proposition 1. For any input S, the conditional distribution $q^\star(\cdot \mid S) = P_t(Y = \cdot \mid S)$ minimizes expected log-loss, so $\mathcal{R}^\star_{\log}(S) = H_t(Y \mid S)$. The chain rule expresses the reduction in optimal log-risk as conditional mutual information:

$$\mathcal{R}^\star_{\log}(Z^g) - \mathcal{R}^\star_{\log}(Z^g, X) = H_t(Y \mid Z^g) - H_t(Y \mid Z^g, X) = I_t(Y; X \mid Z^g). \tag{11}$$

It vanishes exactly when Y and X are conditionally independent given Z^g.

Let Y be a target label, $\mathbf{e}_Y$ its one-hot vector, and $\boldsymbol{\pi}^g, \boldsymbol{\pi}^\ell \in \Delta^{K-1}$ the two predictive distributions for the same target node. For $\boldsymbol{\pi}^\alpha = (1-\alpha)\boldsymbol{\pi}^g + \alpha\boldsymbol{\pi}^\ell$ with $\alpha \in [0,1]$, define the Brier risk as $\mathcal{R}(\boldsymbol{\pi}) = \mathbb{E}\|\mathbf{e}_Y - \boldsymbol{\pi}\|_2^2$, where the expectation is over the target distribution. Expanding the squared norm gives

$$\begin{aligned}\mathcal{R}(\boldsymbol{\pi}^\alpha) = (1-\alpha)\mathcal{R}(\boldsymbol{\pi}^g) + \alpha\mathcal{R}(\boldsymbol{\pi}^\ell) \\ - \alpha(1-\alpha)\mathbb{E}\|\boldsymbol{\pi}^g - \boldsymbol{\pi}^\ell\|_2^2.\end{aligned} \tag{12}$$

Indeed, applying $\|(1-\alpha)\mathbf{a}+\alpha\mathbf{b}\|_2^2 = (1-\alpha)\|\mathbf{a}\|_2^2+\alpha\|\mathbf{b}\|_2^2-\alpha(1-\alpha)\|\mathbf{a}-\mathbf{b}\|_2^2$ to $\mathbf{a} = \mathbf{e}_Y - \boldsymbol{\pi}^g$ and $\mathbf{b} = \mathbf{e}_Y - \boldsymbol{\pi}^\ell$, then taking expectations, proves Eq. 12. For $\alpha > 0$, fusion improves the graph-aware expert whenever

$$(1-\alpha)\mathbb{E}\|\boldsymbol{\pi}^g - \boldsymbol{\pi}^\ell\|_2^2 > \mathcal{R}(\boldsymbol{\pi}^\ell) - \mathcal{R}(\boldsymbol{\pi}^g), \tag{13}$$

obtained by subtracting $\mathcal{R}(\boldsymbol{\pi}^g)$ from both sides of Eq. 12. The disagreement term measures predictive diversity, while the condition identifies when this diversity is large enough to offset the local expert's risk gap. To prove the second part of Proposition 1, set $\Delta = \mathcal{R}(\boldsymbol{\pi}^\ell) - \mathcal{R}(\boldsymbol{\pi}^g) \geq 0$ and $D = \mathbb{E}\|\boldsymbol{\pi}^g - \boldsymbol{\pi}^\ell\|_2^2$. Equation 12 yields

$$\mathcal{R}(\boldsymbol{\pi}^\alpha) - \mathcal{R}(\boldsymbol{\pi}^g) = \alpha(\Delta - D) + \alpha^2 D. \tag{14}$$

If $D \leq \Delta$, the right-hand side is nonnegative for every $\alpha \in [0,1]$. If $D > \Delta$, its minimizer on $[0,1]$ is $\alpha^\star = (D-\Delta)/(2D) \in (0, 1/2]$, and substitution gives the strict risk reduction in Eq. 4.

Together, conditional information identifies attribute signal beyond the graph-aware state, while the Brier condition determines whether the trained graph-free local expert converts that signal into a lower-risk mixture. The correctness decomposition and matched capacity and ensemble controls evaluate how these conditions relate to the predictions of the trained experts.

B Reproduction Details

B.1 Training and inference

Each task and run uses the following independent-training and fusion procedure:

1. Initialize the two experts with disjoint parameters and separate optimizers.
2. At each epoch, compute source and target graph embeddings, form detached entropy-dependent sampling probabilities, and update the graph parameters using $\mathcal{L}^g_{\text{src}} + \lambda\mathcal{L}_{\text{EAM}}$.
3. Update the local MLP once on source cross-entropy with its own optimizer.
4. Store both experts' target logits at each epoch for probability-fusion replay.

The optimizers have disjoint parameter groups. EAM gradients remain within the graph parameter block, and fusion operates on stored predictions after training.

B.2 Graph-aware and graph-free local experts

For the graph-aware expert, let $\widetilde{\mathbf{A}}_d$ be obtained from the processed adjacency matrix $\mathbf{A}_d$ by inserting a unit self-loop only where the diagonal entry is zero. Under the convention that $[\mathbf{A}_d]_{ij} > 0$ sends a message from node j to node i, propagation uses symmetric degree normalization:

$$\mathbf{D}_d = \operatorname{diag}(\widetilde{\mathbf{A}}_d \mathbf{1}_{N_d}), \qquad \mathbf{S}_d = \mathbf{D}_d^{-1/2}\widetilde{\mathbf{A}}_d\mathbf{D}_d^{-1/2}. \tag{15}$$

The A2GNN encoder in Eq. 1 applies P_d propagation steps before one feature transformation and ReLU, followed by one graph classification layer. Thus $P_d = 0$ removes propagation from the feature extractor but retains the graph classifier. The graph-free local expert F_ℓ is a two-layer ReLU MLP applied independently to each row of $\mathbf{X}_d$. The graph-aware and local-expert hidden widths are 128 and 64, respectively; both classification heads return normalized class probabilities.

B.3 Exact EAM implementation

Let $m = \min(1000, N_s, N_t)$ and $R = 5$. For repetition r, source indices are sampled uniformly with replacement and target indices are sampled with replacement from Eq. 5. Sampling weights and graph probabilities used to construct $\mathbf{q}$ are detached; sampled target embeddings retain gradients. We set the source gradient scale to $\gamma = 0$, so sampled source embeddings serve as fixed anchors. Let $\mathbf{S}_r = [\mathbf{s}_{r,1}, \ldots, \mathbf{s}_{r,m}]^\top$ and $\mathbf{T}_r = [\mathbf{t}_{r,1}, \ldots, \mathbf{t}_{r,m}]^\top$ collect the sampled embeddings, and let $\mathbf{U}_r = [\mathbf{S}_r; \mathbf{T}_r] \in \mathbb{R}^{2m \times h_g}$ with $\mathbf{u}_{r,a}$ denoting its ath row. We compute a detached base bandwidth for each repetition and combine five RBF kernels with geometric bandwidths:

$$\bar{b}_r = \frac{1}{2m(2m-1)} \sum_{\substack{a,b=1 \\ a \neq b}}^{2m} \|\mathbf{u}_{r,a} - \mathbf{u}_{r,b}\|_2^2, \quad b_r = \max\{\mathrm{sg}[\bar{b}_r], \epsilon_{\mathrm{mach}}\},$$
$$k_r(\mathbf{u}, \mathbf{v}) = \sum_{j=0}^{4} \exp\left(-\frac{\|\mathbf{u} - \mathbf{v}\|_2^2}{b_r \, 2^{j-2}}\right). \tag{16}$$

Together with Eq. 7, this specifies the exact biased MMD V-statistic used in training. The matched Uniform control sets $q_i = 1/N_t$ and executes this same replacement sampler, bandwidth computation, kernel mixture, and estimator for both source and target domains.

With the fixed floor $\varepsilon = 0.8$, the unnormalized sampling weights lie in $[0.8, 1]$, so the sampling probabilities of any two target nodes differ by a factor of at most 1.25. EAM therefore adjusts participation continuously rather than selecting a hard confidence subset. Replacement draws permit a node to contribute more than once, and averaging repeated estimates exposes the alignment objective to multiple sampled node sets. The floor controls relative participation, while the sample cap and repetition count control the computational budget of the discrepancy estimate.

B.4 Shared settings and family-level search spaces

Tables 3 and 4 list the fixed settings and family-level search spaces.

Table 3: Shared architecture, optimization, and estimator settings.

Component	Globally frozen setting
Experts	Graph hidden width 128; local hidden width 64; one graph feature layer; ReLU; no dropout
Optimization	Adam; local learning-rate scale 1; 150 epochs; no warmup
EAM	Floor 0.8; source gradient scale $\gamma = 0$; sample cap 1000; five replacement draws
Kernel estimator	Biased MMD; five RBF kernels; multiplier 2; no cosine mixture
Fusion	Task-level constant mixture in probability space

The trained bundle varies the graph learning rate, source/target propagation pair, and alignment weight. Every task evaluates the same $3 \times 8 \times 4 = 96$ family-level bundles in Table 4. Thus, the trained search has three task-sensitive dimensions. Architecture and estimator settings are fixed globally; weight decay is fixed within each family as listed in Table 4, not retuned per transfer. The task-level fusion coefficient is evaluated by replaying the stored expert predictions. All transfers use the same coarse candidate set $\alpha_\tau \in \{0.05, 0.1, 0.4, 0.6\}$. Within each transfer, one fixed coefficient combines the graph-aware and graph-free probability vectors for all target nodes.

Table 4: Family-level search spaces for the 96 trained bundles per task.

Family	Graph LR	Propagation pairs (P_s/P_t)	λ	WD
Citation	$\{.005, .01, .02\}$	$0/0, 1/1, 5/5, 10/10$ $0/10, 10/0, 5/10, 10/5$	$\{.1, .3, .5, 1\}$	.001
Airport	$\{.003, .01, .03\}$	$0/0, 1/1, 5/5, 10/10$ $15/15, 0/10, 10/0, 5/10$	$\{.01, .1, 1, 10\}$	.005
Blog	$\{.001, .003, .01\}$	$0/0, 1/1, 5/5, 10/10$ $0/10, 10/0, 5/10, 10/5$	$\{.01, .03, .1, 1\}$	.005
Twitch	$\{.001, .005, .02\}$	$0/0, 1/1, 5/5, 10/10$ $0/10, 10/0, 5/10, 10/5$	$\{.1, .3, .5, 1\}$	.005

B.5 Datasets and preprocessing

All methods read the same processed tensors. Citation uses the supplied dense document attributes; Twitch expands the supplied feature indices into 3,170-dimensional binary vectors; Airport follows the shared 241-dimensional one-hot degree-feature construction; and Blog reads the provided `attrb` matrices. Features retain the numerical scale supplied by the processed data.

Edges retain the directed representation received by the trainer, including stored duplicate edges and self-loops in Airport. Each graph convolution applies standard GCN normalization and inserts missing self-loops. The local expert operates on node attributes independently of `edge_index`; Airport attributes retain their supplied degree-derived entries.

Table 5: Statistics of the processed graphs used by all transfer tasks.

Domain	Family	Nodes	Edges	Features	Classes
BRAZIL	Airport	131	2,148	241	4
EUROPE	Airport	399	11,990	241	4
USA	Airport	1,190	27,198	241	4
Blog1	Blog	2,300	66,942	8,189	6
Blog2	Blog	2,896	107,672	8,189	6
ACMv9	Citation	9,360	31,112	6,775	5
Citationv1	Citation	8,935	30,196	6,775	5
DBLPv7	Citation	5,484	16,234	6,775	5
DE	Twitch	9,498	153,138	3,170	2
EN	Twitch	7,126	35,324	3,170	2

B.6 Evaluation and statistical analysis

Macro-F1 is the primary metric. All tabulated Macro- and Micro-F1 values are reported as percentages. Micro@Macro evaluates Micro-F1 at the Macro-best epoch, while independent Micro-best may select a different epoch. Macro-best and independent Micro-best coincide in 58 of 80 selected task–run trajectories and differ in 22. Family and Full-16 scores are equal-task averages. Controlled contrasts pair task and run. The confidence intervals for aggregate expert and capacity comparisons use 10,000 two-level paired-bootstrap replicates that first resample tasks and then paired runs within each task. We take the 2.5th and 97.5th percentiles of the resulting paired-difference distribution.

C Efficiency Analysis

C.1 Computational complexity

Let h_g and h_ℓ denote the graph and local hidden widths. Graph propagation and linear transformations cost $O(P_d|E_d|D + |E_d|h_g + N_d D h_g + N_d h_g K)$ in domain d. The local expert costs $O(N_d D h_\ell + N_d h_\ell K)$. Each of the R EAM repetitions forms pairwise kernels over $2m$ embeddings, giving $O(Rm^2 h_g)$ time and $O(m^2)$ auxiliary memory. EAM is a training-time operation; inference comprises the graph-aware expert, one local-expert forward pass, and probability interpolation.

C.2 FINAL-MODEL EFFICIENCY

Table 6 measures final-model training and inference under one matched A→C configuration on an RTX 4090 D GPU, separately from the one-time hyperparameter-search cost.

Table 6: Final-model efficiency under a common A→C profiling setup.

Variant	Parameters	s/epoch	Peak MiB	Inference ms
Graph-only	867,973	0.710	1660	3.753
Parameter-matched graph	1,301,957	0.813	1854	3.756
Graph+Graph	1,735,946	1.421	1660	7.586
Graph+Local (EviGDA)	1,301,962	0.726	1660	4.105

Relative to Graph-only, the complete method adds 50.0% trainable parameters while increasing measured training time by 2.3% and inference latency by 9.4%; peak memory remains unchanged in this profile. At nearly identical parameter count, it trains 10.7% faster than the widened parameter-matched graph control. It also reduces training and inference time by 48.9% and 45.9%, respectively, compared with Graph+Graph. The graph-free local expert therefore provides heterogeneous evidence at substantially lower execution cost than a second graph expert.

The operation types help explain this difference between parameter and runtime overhead. The local expert applies two node-wise dense transformations without neighborhood aggregation or pairwise alignment kernels. Adding its parameters therefore does not duplicate the graph expert's propagation and EAM workload. At inference, combining the two probability vectors requires only $O(N_t K)$ arithmetic operations, with no additional graph traversal. This separation is consistent with the modest measured latency increase despite the larger parameter count.

D ADDITIONAL EXPERIMENT RESULTS

Consistency across runs. Across leave-one-run evaluations, the resulting operating points retain 99.48% of the Full-16 Macro-F1 score obtained using all five runs. Family-level scores remain similarly consistent, indicating stable responses across repeated runs.

D.1 COMPLETE MICRO-F1 COMPARISON

Table 7 complements the main Macro-F1 comparison with Micro-F1 on all 16 transfers. The same task ordering and method grouping are retained so that class-balanced and frequency-weighted performance can be examined together under a common transfer setting.

For single-label node classification, Micro-F1 pools true positives, false positives, and false negatives over classes and equals the fraction of correctly classified nodes. Macro-F1 instead averages class-specific F1 scores with equal class weights. The two metrics therefore emphasize different aspects of prediction when class frequencies are unequal: Micro-F1 summarizes node-level correctness, while Macro-F1 gives each class equal influence on the reported score.

Cross-metric consistency. EviGDA obtains the highest mean Micro-F1 on 15 of 16 transfers, including all Airport, Blog, and Twitch directions. On Citation, it reaches 83.22 on A→C and 79.28 on C→D; DGSDA retains the strongest D→C result at 82.55. The broad agreement with the main Macro-F1 comparison shows that the gains extend to frequency-weighted node prediction, rather than appearing only under equal weighting of classes.

Family-level behavior. The gains over the strongest competing Micro-F1 results are 5.73 and 3.12 points on B1→B2 and B2→B1, respectively. Airport also shows clear improvements, including 4.81 points on U→B and 3.90 on B→E. Twitch gains are smaller: EviGDA reaches 60.80 on DE→EN and 65.82 on EN→DE. These differences agree with the main paper's family-dependent gains: additional graph-free predictions are more useful on some transfers than on others.

Table 7: Complete Micro-F1 comparison (%) across the 16 transfer tasks. Entries report mean ± sample standard deviation; bold and underline denote the best and second-best results.

Method	A→C	A→D	C→A	C→D	D→A	D→C	B1→B2	B2→B1
Source-only								
GCN (ICLR'17)	70.64 ± 1.51	66.18 ± 0.69	66.59 ± 0.94	71.04 ± 0.80	59.10 ± 0.71	68.59 ± 2.37	33.31 ± 1.97	31.86 ± 1.50
Graph domain adaptation								
UDAGCN (WWW'20)	80.11 ± 1.08	74.43 ± 1.05	72.90 ± 1.48	77.88 ± 1.29	65.29 ± 2.17	76.18 ± 1.17	34.99 ± 2.94	33.44 ± 1.52
GRADE (AAAI'23)	72.53 ± 1.66	67.73 ± 2.26	67.76 ± 0.36	72.86 ± 1.00	63.07 ± 0.36	69.45 ± 1.06	46.66 ± 2.32	46.06 ± 1.26
PairAlign (ICML'24)	61.37 ± 2.09	60.08 ± 2.51	55.42 ± 1.67	61.41 ± 2.46	53.00 ± 1.00	58.16 ± 2.41	41.66 ± 1.63	44.00 ± 0.90
GraphAlign (KDD'24)	69.31 ± 1.13	65.88 ± 1.46	64.18 ± 0.82	69.38 ± 1.24	59.46 ± 0.88	65.17 ± 0.98	39.73 ± 1.92	43.78 ± 1.04
A2GNN-MMD (AAAI'24)	80.82 ± 0.35	76.05 ± 0.26	75.68 ± 0.32	76.39 ± 1.12	73.59 ± 0.45	80.14 ± 0.59	40.99 ± 1.44	42.94 ± 1.59
TDSS-RW (AAAI'25)	81.81 ± 0.40	77.55 ± 0.51	74.59 ± 0.45	76.55 ± 0.41	73.15 ± 0.58	80.49 ± 0.05	45.00 ± 0.92	43.99 ± 0.90
GAA (ICLR'25)	75.56 ± 1.57	68.17 ± 1.02	74.77 ± 0.78	68.25 ± 1.04	68.97 ± 0.91	71.21 ± 0.59	46.11 ± 1.43	47.77 ± 1.70
DGSDA (ICML'25)	82.88 ± 0.30	76.15 ± 0.68	74.70 ± 0.22	77.67 ± 0.35	73.20 ± 0.36	**82.55** ± 0.49	38.17 ± 1.46	41.98 ± 1.38
HGDA (ICML'25)	71.36 ± 1.09	72.54 ± 1.77	66.62 ± 1.91	73.05 ± 1.57	67.57 ± 1.74	74.99 ± 1.20	42.47 ± 0.79	43.96 ± 1.04
ADAlign (ICLR'26)	81.54 ± 0.50	77.59 ± 0.54	75.96 ± 0.37	78.22 ± 1.23	70.70 ± 1.14	78.43 ± 1.18	44.85 ± 1.59	44.59 ± 1.40
DiffGDA (ICLR'26)	79.55 ± 1.20	74.11 ± 1.36	73.15 ± 0.94	77.41 ± 0.79	68.67 ± 1.35	77.22 ± 0.46	42.10 ± 1.32	41.10 ± 1.74
DFT (KDD'26)	74.45 ± 1.81	72.84 ± 0.49	68.35 ± 0.79	73.09 ± 1.00	66.53 ± 1.34	78.20 ± 1.59	48.31 ± 1.42	51.34 ± 0.93
EviGDA (Ours)	**83.22** ± 0.14	**78.50** ± 0.77	**76.06** ± 0.19	**79.28** ± 0.23	**73.94** ± 0.27	82.18 ± 0.40	**54.04** ± 1.73	**54.46** ± 0.63

Method	U→B	U→E	B→U	B→E	E→U	E→B	DE→EN	EN→DE
Source-only								
GCN (ICLR'17)	50.38 ± 1.05	37.69 ± 1.27	44.08 ± 0.47	36.39 ± 1.43	45.36 ± 0.26	42.14 ± 0.98	56.85 ± 0.42	59.78 ± 1.09
Graph domain adaptation								
UDAGCN (WWW'20)	62.25 ± 0.85	44.35 ± 0.93	41.82 ± 0.66	51.62 ± 0.93	42.18 ± 0.64	61.37 ± 1.16	58.45 ± 0.64	63.15 ± 0.53
GRADE (AAAI'23)	62.44 ± 1.51	48.87 ± 1.81	42.20 ± 1.64	55.44 ± 1.14	46.47 ± 1.77	70.08 ± 1.74	59.25 ± 0.34	63.99 ± 0.31
PairAlign (ICML'24)	70.00 ± 1.76	41.45 ± 0.65	47.41 ± 1.26	40.45 ± 1.21	43.38 ± 1.17	49.62 ± 1.29	60.28 ± 0.40	65.23 ± 0.33
GraphAlign (KDD'24)	69.08 ± 0.87	57.44 ± 0.80	49.82 ± 0.51	55.75 ± 0.85	52.12 ± 0.47	69.66 ± 0.60	54.67 ± 1.20	59.78 ± 1.86
A2GNN-MMD (AAAI'24)	61.98 ± 0.72	51.63 ± 1.42	44.62 ± 1.48	54.04 ± 0.29	44.13 ± 0.86	66.26 ± 1.35	57.24 ± 0.37	59.74 ± 1.25
TDSS-RW (AAAI'25)	75.31 ± 2.25	45.25 ± 0.34	56.10 ± 6.49	44.35 ± 2.99	51.24 ± 3.58	62.93 ± 0.05	56.45 ± 0.50	57.71 ± 0.69
GAA (ICLR'25)	75.42 ± 1.39	54.74 ± 1.62	55.62 ± 1.49	54.84 ± 1.08	50.82 ± 1.23	74.20 ± 1.77	55.38 ± 1.36	60.85 ± 0.99
DGSDA (ICML'25)	63.97 ± 1.47	53.88 ± 1.19	57.71 ± 0.22	53.13 ± 0.77	52.06 ± 0.62	71.45 ± 1.16	60.35 ± 0.14	63.43 ± 0.65
HGDA (ICML'25)	75.40 ± 0.78	44.00 ± 0.50	57.80 ± 0.64	53.18 ± 0.28	47.61 ± 0.20	66.05 ± 0.39	56.62 ± 1.02	60.67 ± 1.06
ADAlign (ICLR'26)	75.57 ± 1.23	53.73 ± 1.06	49.73 ± 1.27	55.69 ± 1.13	51.04 ± 1.03	69.16 ± 1.20	59.76 ± 0.28	64.58 ± 0.44
DiffGDA (ICLR'26)	76.26 ± 1.06	48.07 ± 1.47	45.70 ± 1.25	45.92 ± 0.85	47.76 ± 1.20	65.34 ± 1.05	56.37 ± 0.30	59.66 ± 0.83
DFT (KDD'26)	76.56 ± 1.02	51.33 ± 0.75	53.24 ± 1.21	52.78 ± 1.55	50.03 ± 0.89	69.31 ± 1.55	60.73 ± 0.60	64.73 ± 1.16
EviGDA (Ours)	**81.37** ± 0.42	**58.25** ± 0.76	**58.32** ± 0.39	**59.65** ± 0.18	**52.69** ± 0.41	**76.64** ± 1.16	**60.80** ± 0.13	**65.82** ± 0.57

D.2 Probability Quality

The correctness decomposition in Figure 2(a) is complemented by probability-quality statistics. Mean multiclass Brier scores are 0.5331 for Graph-only, 0.6512 for Local-only, and 0.5037 after fusion. Mean squared expert disagreement is 0.2695, the probability-mixture diversity credit is 0.0148, and the decomposition residual is below 3×10^{-9} in magnitude. Fusion therefore reduces Brier risk by 0.0293 relative to Graph-only even though Local-only has higher standalone risk. This pattern is characteristic of complementary experts: the local probabilities contribute useful class mass on disagreement nodes, while the graph-dominant mixture preserves the stronger graph-aware expert on the remaining nodes. The near-zero residual also confirms the numerical agreement between the observed mixture risk and Eq. 12 for the trained expert pair.

D.3 Matched EAM control

The matched Uniform variant uses the same exact sampler, sample count, kernel, bandwidth rule, and gradient routes as EAM; uniform target mass $q_i = 1/N_t$ is the controlled intervention. Across the 16 transfer tasks, EAM improves the fused Macro-F1 endpoint by 0.24 points on average. The average gains by family are $+0.35$ (Citation), $+0.06$ (Airport), $+0.59$ (Blog), and $+0.08$ (Twitch), all measured in Macro-F1 points relative to the matched control.

Table 8: Per-task EAM gains over matched Uniform sampling (Macro-F1 percentage points).

Task	Δ Macro-F1	Task	Δ Macro-F1
A→C	+0.03	B→U	+0.02
A→D	+1.16	E→U	−0.03
C→A	+0.17	U→B	+0.17
C→D	+0.43	U→E	+0.07
D→A	−0.04	B1→B2	+0.39
D→C	+0.37	B2→B1	+0.80
B→E	−0.03	DE→EN	+0.20
E→B	+0.17	EN→DE	−0.04

The response follows the quality and coverage of confident target evidence. Representative improvements include A→D (+1.16 points), B2→B1 (+0.80), C→D (+0.43), and D→C (+0.37). In these directions, low-entropy predictions repeatedly provide concentrated anchors for the empirical alignment measure. More moderate gains arise when the uniform and entropy-tilted samplers already cover similar target regions. Because the kernel, sample count, and gradient route are matched, the observed differences directly reflect how entropy-guided participation changes the target evidence presented to the same kernel discrepancy estimator.

D.4 Controlled topology-shift stress test

The main paper reports graph-expert degradation and local-exclusive evidence. Here we verify that the intervention reduces target homophily and show the corresponding fusion response. A→C, U→B, B1→B2, and EN→DE each use six degree-preserving rewiring levels and five runs (120 trajectories); the fusion coefficient remains fixed across severities.

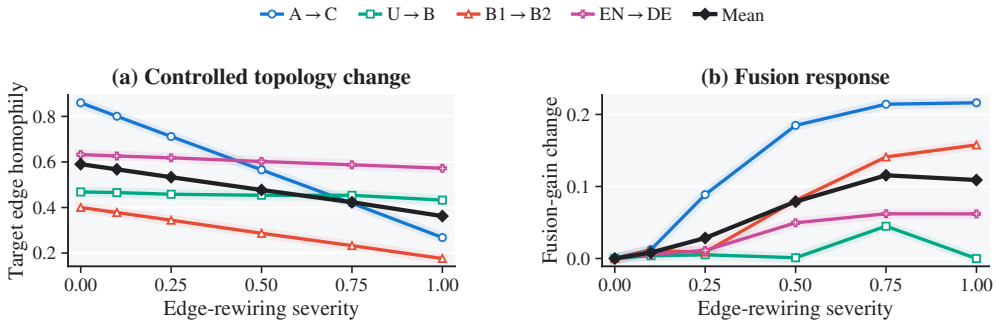


Figure 6: **Supplementary topology intervention.** Degree-preserving rewiring lowers target homophily, while the fusion response varies by graph family. The black line is the equal-task mean.

Severity zero reproduces the reference endpoint, and local-expert logits have identical hashes across severities. Table 9 summarizes the maximum-severity changes. The largest response occurs on A→C: graph damage reaches 69.86 points, local-exclusive correctness rises by 33.38 points, and the fusion gain increases by 21.62 points. B1→B2 and EN→DE form intermediate regimes, whereas U→B remains nearly stable. These trajectories expose a graded relationship between neighborhood degradation and the value of graph-free local evidence rather than a binary task split.

Table 9: Endpoint changes at maximum target-topology corruption (percentage points).

Transfer	Graph damage	Local-exclusive gain	Fusion-gain change
A→C	69.86	33.38	21.62
U→B	0.76	0.76	−0.00
B1→B2	27.10	14.27	15.77
EN→DE	10.36	8.12	6.19
Equal-task mean	27.02	14.13	10.90

Table 9 reports the primary maximum-severity endpoints. A complementary fixed-reference-epoch replay yields mean graph-damage, local-exclusive, and fusion-gain changes of 31.29, 11.67, and 1.91 points, respectively. Both readouts preserve the same ordering: transfers with stronger graph-expert degradation expose more local-exclusive evidence and larger gains from combining the two experts.

D.5 Hyperparameter response

The search evaluates 96 trained bundles per task and replays the task-level coefficient on the shared coarse grid $\alpha_\tau \in \{0.05, 0.1, 0.4, 0.6\}$. Coefficient replay over stored expert predictions is parameter-free. Figure 7 profiles each search dimension while optimizing over the remaining dimensions; lower gaps indicate wider high-performing regions across the transfer tasks.

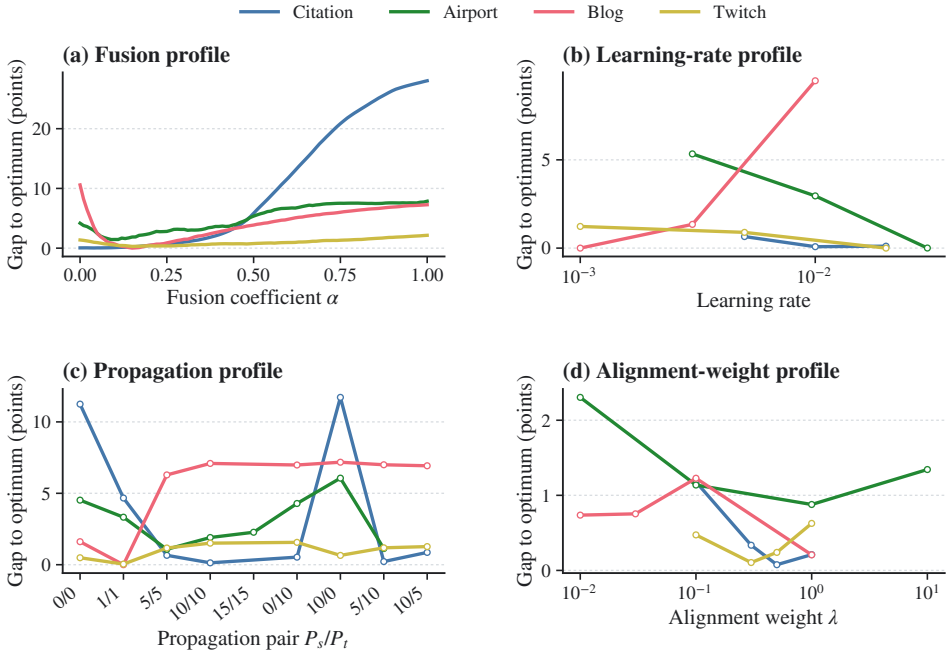


Figure 7: **Family-level hyperparameter profiles.** Each point reports the mean gap to the task optimum after profiling over the remaining search dimensions; lower is better.

The following analysis relates each parameter response to its role in graph transfer.

Study on the fusion coefficient. The coefficient controls the balance between graph-aware and graph-free local evidence. The shared grid covers graph-dominant fusion, light local-expert participation, and progressively stronger local-expert contributions. Citation generally favors the graph-dominant end, whereas Airport, Blog, and Twitch contain transfers that benefit from larger local weights. The preferred point may differ between transfer scenarios, but within a scenario it is one scalar shared by all nodes. This controlled response supports a compact task-level probability mixture.

Study on propagation depth. Propagation determines how strongly each domain incorporates neighborhood information before alignment. Symmetric and asymmetric pairs form several competitive operating regions, with their ordering varying across families. This agrees with the premise that source and target graphs can require different degrees of structural smoothing and motivates evaluating source and target propagation together as a paired configuration.

Study on learning rate and alignment weight. Learning-rate profiles retain clear favorable regions for each family. The alignment weight exhibits the expected balance: increasing it strengthens cross-domain matching until source discrimination and alignment reach an effective trade-off. Across both dimensions, the response curves identify family-level differences in the balance between source fitting and target alignment.