Adaptive Attribute Completion with Representation Space for Incomplete Graph Domain Adaptation

Niya Yang[1,2] **Di Jin**[1] **Zhizhi Yu**[1*] **Liang Yang**[3] **Yuxiao Huang**[4]
Dongxiao He[1] **Jianguo Wei**[1]
[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]School of Information Security Technology, Jilin Police College, Changchun, China
[3]Hebei Province Key Laboratory of Big Data Calculation,
School of Artificial Intelligence, Hebei University of Technology, Tianjin, China
[4]Data Scienc, George Washington University, Washington, D.C., USA
{niya_yang1,jindi,yuzhizhi,hedongxiao,jianguo}@tju.edu.cn
yangliang@vip.qq.com, yuxiaohuang@gwu.edu

Abstract

Graph Domain Adaptation (GDA) has emerged as an effective technique for transferring knowledge from a label-rich source graph to an unlabeled target graph by mitigating cross-graph discrepancies. However, in real-world scenarios, the attributes of partial nodes may be unobserved (i.e., missing) due to factors such as privacy protection. This inevitably exacerbates the distribution discrepancy of nodes between graph domains, making it more challenging to transfer effective knowledge to the target graph. To tackle this issue, we propose an adaptive attribute completion approach with representation space for incomplete graph domain adaptation, namely AC-GDA. It mutually assists the attribute completion and dynamically optimizes cross-graph trustworthy embeddings in an iterative manner, promoting positive knowledge migration while completing attributes. Specifically, we first propose a joint completion mechanism guided by topology similarity, which leverages cross-graph trustworthy attribute embeddings with imputation weights to adaptively complete missing node attributes. We then enhance node embeddings with label information to provide supervision for completed nodes, making nodes of the same category distribute consistently across graph domains. Furthermore, we introduce the complement confidence score to dynamically adjust the scope of trustworthy embeddings, so as to improve the quality of subsequent cross-graph imputation. Extensive experiments on a range of cross-graph tasks validate the effectiveness of AC-GDA.

1 Introduction

Graph-structured data effectively captures diverse entities and the complex relationships among them, making it highly valuable for extracting meaningful insights in real-world scenarios. Recently, Graph Neural Networks (GNNs) [1] have played a pivotal role in graph-related applications by enabling effective representation learning and facilitating downstream analysis tasks. However, when applied to unlabeled graphs, existing GNNs suffer a noticeable decline in performance due to lack of supervision. To eliminate the reliance on manual label annotations, Graph Domain Adaptation (GDA) has emerged as a powerful technique for transferring abundant expert knowledge from a label-rich source graph to an unlabeled target graph. To data, GDA methods foster significant improvements on various practical applications, such as recommendation [2], social analysis [3], traffic forecasting [4], and video application [5].

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

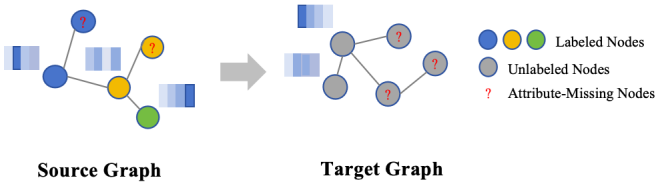


Figure 1: An illustrative example of transferring label knowledge between attribute-missing graphs.

Most existing GDA methods [6, 7, 8] adopt a two-stage paradigm that first learns node embeddings via GNNs [9], and subsequently aligns the embedding distributions of source and target graphs. Consequently, they rely on the implicit assumption of a complete graph domain, in which node attributes along the graph topology are fully observed to support effective node representation learning. This assumption, however, is frequently violated in practice, since many real-world cross-graph tasks suffer from attribute incompleteness. Taking e-commerce recommendation in Figure 1 as an example, certain user attributes may be unavailable due to technical limitations, human error, or restricted access stemming from data confidentiality and privacy concerns. Such incomplete attributes disrupt the learning of node representations within e-commerce and hamper the effective transferring of label knowledge across e-commerce platforms, leading to suboptimal good recommendation. Despite this challenge, to the best of our knowledge, existing GDA methods have not been specifically tailored for graphs with missing attributes. This observation naturally leads to a critical question: is it possible to transfer label knowledge from a source graph with incomplete attributes to a target graph that is likewise attribute-missing, while coping with substantial distributional shifts across domains? Here we define this practical scenario as domain adaptation for attribute-missing graphs.

Nevertheless, it is technically challenging to design an effective domain adaptive architecture for attribute-missing graph. In particular, two fundamental issues need to be addressed. **Issue 1: How to perform attribute completion for both source and target graphs?** Missing attributes pose significant challenges to node representation learning. Traditional attribute completion methods are typically confined to knowledge within a graph [10], potentially overlooking pivotal cross-graph knowledge that is crucial for effective imputation. This limitation becomes more pronounced in the target graph, which is entirely unlabeled and lacks the capacity to bootstrap attribute completion on its own. Therefore, it is essential to leverage cross-graph expert knowledge to assist in attribute completion within both source and target graphs. **Issue 2: How to ensure the quality of completed attributes to facilitate effective label knowledge transfer?** Imputed attributes may contain semantic biases, which might exacerbate distributional discrepancies between domains and hinder the effectiveness of GDA models in transferring label knowledge. Thus, it is crucial to design an adaptive attribute completion mechanism that not only refines the quality of the completed attributes, but also promotes positive and reliable knowledge transfer across graph domains.

To address the aforementioned issues, we propose an adaptive **A**ttribute **C**ompletion approach with representation space for incomplete **G**raph **D**omain **A**daptation, termed **AC-GDA**. It innovatively integrates attribute completion and knowledge transfer in an unified framework, thereby mitigating the distribution discrepancy between graphs due to the biases of complementary attributes. Specifically, we first encode graph topology and attribute-observed information to obtain comprehensive and trustworthy node embeddings, that is, topology embedding and attribute-observed embedding. We then design a joint completion mechanism guided by topology similarity, which utilizes cross-graph trustworthy attribute embeddings with imputation weights to adaptively complete attribute-missing nodes. To further mitigate semantic bias and improve the quality of cross-graph attribute completion, we define a complement confidence score that dynamically expands the scope of trustworthy embeddings for subsequent cross-graph imputation in an end-to-end fashion. Finally, we enhance node embeddings with optimized label to ensure that nodes of the same category distribute consistently between graphs, facilitating positive label knowledge migration.

The main contributions are summarized as follows:

- We find a significant amount of cross-graph tasks where node attributes are partially missing due to factors such as privacy protection, which renders existing graph domain adaptation methods inapplicable.
- We propose AC-GDA, which integrates cross-graph trustworthy embedding imputation and knowledge migration in a unified manner. This approach significantly mitigates the

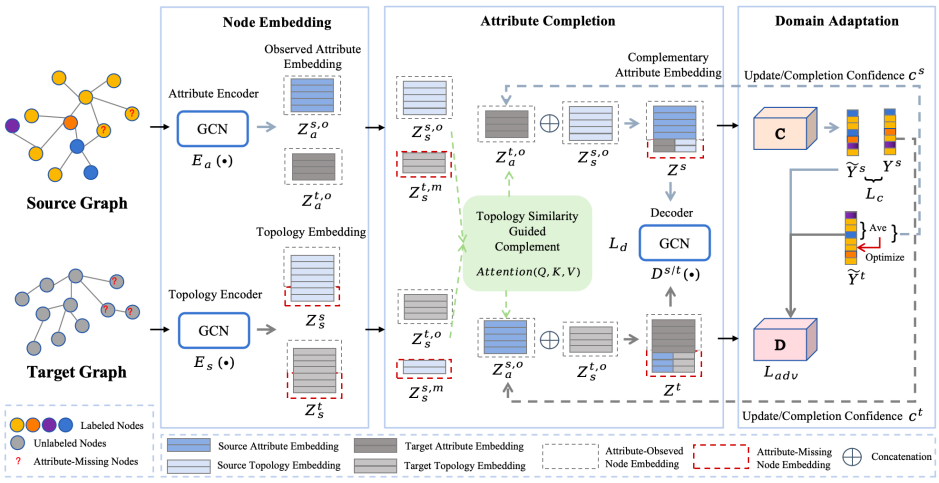


Figure 2: The architecture of AC-GDA. We first learn the attribute-observed embedding and the topology embedding in source and target graphs. Then, we adaptively complete attribute-missing nodes with cross-graph attribute-trustworthy embedding and topology embedding within the graph. Finally, we enhance node embedding with labels to promote positive label knowledge.

distribution discrepancy between graph domains caused by biases in attribute completion, thereby ensuring effective domain adaptation.

- Empirical evaluation on various cross-graph datasets demonstrates the effectiveness of our AC-GDA over existing state-of-the-art methods.

2 Preliminaries

Notations. Given a graph $G = (V, A, X, Y)$, where $V = \{v_1, v_2, ...v_N\}$ is the set of nodes, $A = [a_{ij}] \in \mathbb{R}^{N \times N}$, $X = [x_{ij}] \in \mathbb{R}^{N \times F}$ and $Y = [y_{ij}] \in \mathbb{R}^{N \times C}$ denote adjacency matrix, node feature matrix and label matrix, respectively. $N = |V|$ is the size of node set, F is the dimension of node feature, and C is the number of class. If there is an edge between node v_i and node v_j, $a_{ij} = 1$, otherwise $a_{ij} = 0$. An attribute-missing graph refers to a setting where certain attribute values x_{ij} in the adjacency matrix X are missing, we further define $V^o = \{v_1^o, v_2^o, ..., v_{N^o}^o\}$ and $V^m = \{v_1^m, v_2^m, ..., v_{N^m}^m\}$ to be the set of attribute-observed nodes and attribute-missing nodes, respectively. Accordingly, $V = V^o \cup V^m$, $V^o \cap V^m = \emptyset$, and $N = N^o + N^m$.

Problem Definition. We focus on the downstream task of cross-graph node classification to evaluate our AC-GDA. There are a labeled attribute-missing source graph $G^s = (V^s, A^s, X^s, Y^s)$ and an unlabeled attribute-missing target graph $G^t = (V^t, A^t, X^t)$, our goal is to predict Y^t by jointly learning on source and target graphs with missing attributes.

3 Methodology

3.1 Node Embedding Modeling

To fully leverage all the trustworthy information (i.e., the observed attribute and topology information in both source and target graphs) to complete attributes while achieving the domain alignment. A Pseudo-Siamese architecture [11] is introduced to capture topology and attribute information in source and target graphs, respectively, denoted as topology encoder $E_s^{s/t}(\cdot)$ and attribute encoder $E_a^{s/t}(\cdot)$, respectively. Formally, given a source attribute-observed subgraph $G^{s,o} = \left(X^{s,o}, \hat{A}^{s,o}\right)$ and a target attribute-observed subgraph $G^{t,o} = \left(X^{t,o}, \hat{A}^{t,o}\right)$, where $X^{s/t,o} \in \mathbb{R}^{N^{s/t,o} \times F}$ and $\hat{A}^{s/t,o} \in \mathbb{R}^{N^{s/t,o} \times N^{s/t,o}}$ denote observed feature matrix and normalized adjacency matrix. Then, we fed them into attribute encoder $E_a^{s/t,o}(\cdot)$, and obtain the l-th source attribute-observed node

embeddings $Z_a^{s,o(l)}$ and source attribute-observed node embeddings $Z_a^{t,o(l)}$, respectively:

$$Z_a^{s/t,o(l)} = \mathrm{ReLU}\left(\hat{\mathrm{A}}^{\mathrm{s/t,o}}\mathrm{Z}_{\mathrm{a}}^{\mathrm{s/t,o(l-1)}}\mathrm{W}_{\mathrm{a}}^{\mathrm{s/t,o(l)}}\right), \tag{1}$$

where $Z_a^{s/t,o(0)} = X^{s/t,o}$, $W_a^{s/t,o(l)}$ denotes the parameters of the l-th layer, and $\mathrm{ReLU}(\cdot)$ denotes the activation function. Similarly, a source structural graph $G^{s,s} = \left(I, \hat{A}^s\right)$ and target structural graph $G^{t,s} = \left(I, \hat{A}^t\right)$ are fed into topology encoder $E_s(\cdot)$, respectively, where $I \in \mathbb{R}^{N^{s/t} \times N^{s/t}}$ is identity matrix, and $\hat{A}^{s/t} \in \mathbb{R}^{N^{s/t} \times N^{s/t}}$ denotes the normalized adjacency matrix. The l-th source node embeddings $Z_s^{s(l)}$ and target node embeddings $Z_s^{t(l)}$ can be denoted as:

$$Z_s^{s/t(l)} = \mathrm{ReLU}\left(\hat{\mathrm{A}}^{\mathrm{s/t}}\mathrm{Z}_{\mathrm{s}}^{\mathrm{s/t(l-1)}}\mathrm{W}_{\mathrm{s}}^{\mathrm{s/t(l)}}\right), \tag{2}$$

where $Z_s^{s/t(0)} = I$, $W_s^{s/t(l)}$ denotes the topology encoder parameters of the l-th layer.

3.2 Analysis of Attribute-Topology Consistency

Motivated by the intrinsic consistency and complementarity between node attributes and graph topology [12], and assuming the topological structures of both source and target graphs are complete and reliable, we posit that nodes with similar topological patterns tend to exhibit similar attribute embeddings. We term this assumption the Attribute–Topology Similarity Consistency Hypothesis.

Consistency Lemma. Given source node $u \in V^s$ and target node $v \in V^t$, we assume that attribute embedding similarity $S_a(u, v)$ and topology embedding similarity $S_s(u, v)$ are positively correlated (i.e., high topology similarity implies high attribute similarity, and vice versa). Formally, there exists a function $F \to [0, 1]$ as:

$$S_a(u, v) = F(S_s(u, v)) + \delta_{uv}, \quad \delta_{uv} \sim \mathcal{N}(0, \sigma^2).$$

To rigorously examine our hypothesis that topology and attribute similarity are positively correlated, we randomly sample $1{,}000$ unordered node pairs from the ArnetMiner dataset under a challenging setting where 60% of the node attributes are missing, and provide the following qualitative analysis and the quantitative analysis, respectively.

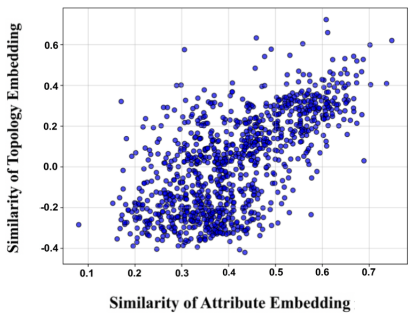


Figure 3: Attribute similarity and topology similarity between pairs of nodes in cross-graph task.

Qualitative Analysis. For each pair of node v_i and node v_j, we compute two quantities, that is, *topology similarity* is measured in the learned topology embedding space, and *attribute similarity* is computed in the attribute embedding space after imputation. As illustrated in Figure.3, the resulting scatter plot exhibits a clear upward trend, indicating a strong positive relationship between the attribute similarity and topology similarity.

Quantitative Analysis. Computing the Pearson correlation coefficient r and its corresponding significance level p, with following the standard methodology recommended in [13]. Across the above $1{,}000$ node pairs from the ArnetMiner, the correlation coefficient remains consistently high ($r \in [0.7251, 0.8798]$), indicating a strong linear relationship between topology and attribute similarity. Moreover, the associated p-values are consistently below 0.05, close to zero, rejecting the null hypothesis of independence at a 95% confidence level. In short, these results provide statistically compelling evidence that topology similarity and attribute similarity are strongly and positively correlated on attribute-missing graph, thereby supporting our proposed key assumption.

3.3 Attribute Completion Modeling

Attribute completion consists of two components: joint completion and dynamic optimization.

Joint Completion. Considering that the target graph is completely unlabeled and unsupervised, classical attribute completion methods, including zero-filling, parameter-filling, and graph-aware filling [14, 15, 16], may introduce a significant amount of irrelevant information, exacerbating the distribution discrepancy between source and target graphs.

According to attribute-topology similarity consistency, we propose a joint completion mechanism guided by topology similarity, which leverages cross-graph attribute-trustworthy embedding with imputation weights to adaptively complete attribute-missing nodes, except for topology complement within the graph in the representation space. Specifically, when completing the target attribute-missing node $v_i^{t,m}$, we select its topology embedding $z_{s,i}^t$, and compute the similarity with cross-graph all topology embeddings Z_s^s, denoted as cross-graph imputation weight set $S_i^s\,(i = 1, 2, ..., N^s)$. Then, we integrate source observed attribute embedding $z_{a,j}^{s,o}\,(j = 1, 2, ..., N^o)$ with corresponding imputation weight to fill the attribute embedding $z_{a,i}^{t,m}$. Mathematically, the above cross-graph attribute completion process can be described based on attention mechanism [17]:

$$z_{cross,i} = Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{D}}\right)V, \tag{3}$$

where Q is the topology embedding $z_{s,i}^t$, K is cross-graph topology embedding $z_{s,j}^s$, V is the corresponding attribute embedding $z_{a,j}^{s,o}$, and D is the dimension.

Meanwhile, inspired by [11], the in-domain topology embedding of the node $v_i^{s,m}$ in the graph should be fully considered in the complementary process. We intuitively concatenate topology embedding $z_s^{t,m}$ of attribute-missing node set V^m and attribute embedding $z_a^{t,o}$ of attribute-observed node set V^o to form complete node representation, and fusing adjacency relations $\hat{A}^t$ to bridge the gap between attribute embedding and topology embedding:

$$z_{intra} = \hat{A}^t\left(Z_a^{t,o}, Z_s^{t,m}\right). \tag{4}$$

Therefore, joint completion for target attribute-missing node embedding $z_{a,i}^{t,m}$ can be defined as:

$$Z_{a,i}^{t,m} = \lambda z_{cross,i} + (1 - \lambda)\, z_{intra,i}, \tag{5}$$

where λ is a balance parameter.

Similarly, we select target attribute-observed node embeddings $Z_{a,i}^{t,o}\,(i = 1, 2, ..., N^t)$ with corresponding imputation weight S_i^t, and combine them with source topology embeddings $Z_s^{s,m}$ to initialize source node embeddings $Z_a^{s,m}$.

Dynamic Optimization. Since we leverage cross-graph attribute-observed embeddings to complete attribute-missing nodes, the quality of attribute completion largely depends on the availability of cross-graph observed attributes. When the number of missing attributes is excessively high (i.e., there are too few observed attributes to support cross-graph completion), semantic deviations in the complementary attributes may increase, thereby compromising the overall performance of attribute completion across graphs.

To this end, we design a dynamic adjustment of trustworthy embedding strategy to divert trusted complementary nodes V^m to the set of attribute-observed nodes V^o. Specifically, a completion confidence score is defined to evaluate the quality of complementary nodes, inspired by [18], the completion confidence c_i of node v_i and confidence threshold α can be defined as follow:

$$c_i = \frac{\sum_{v_j \in S_{i,k}^s} \|z_i - z_j\|_2}{\sum_{v_j \in S_{i,k}^d} \|z_i - z_j\|_2}, \tag{6}$$

where $S_{i,k}^d$ represents the nearest k nodes of different classes surrounded node v_i, and $S_{i,k}^s$ represents the nearest k nodes of the same class. Here, $c_i > \alpha\,(0 < \alpha < 1)$ indicates complementary node v_i being closer to the same class of nodes, which is high quality complementary and will be transferred into attribute-observed node V^e and used for the next round of cross-graph attribute completion. Conversely, $c_i \leq \alpha$ indicates complementary node v_i being closer to the different classes of nodes, which is low quality complementary and will not be transferred into attribute-observed nodes V^o. It is worth noting that the complementary confidence of the node can be computed on the assumption

that all the nodes have labels. In the cross-graph task, source nodes have trustworthy labels, the completely unlabeled target nodes only rely on pseudo-labels generated by the classifier, we consider employing the average category of k-Nearest Neighbors to optimize the pseudo-labels $\hat{Y}$.

Due to both joint complement and dynamic optimization are performed for attribute complement in representation space, it is essential to ensure that the complementary attributes are consistent with the original graph semantics. Inspired by [19, 11], we decode source and target node embedding $Z^{s/t}$ with $\hat{A}^{s/t}$ in representation space, which is formulated as:

$$Z^{s/t(l)} = \mathrm{ReLU}\left(\hat{\mathrm{A}}^{\mathrm{s/t}}\mathrm{Z}^{\mathrm{s/t(l-1)}}\right)\hat{\mathrm{W}}^{\mathrm{s/t(l)}}, \tag{7}$$

where $\hat{W}^{s/t(l)}$ indicates the l-th layer parameters of decoder. $Z^{s/t(0)}$ and $Z^{s/t(2)}$ denote the decoded embeddings $Z^{s/t}$ and the decoded attribute matrix $\hat{X}^{s/t}$, respectively.

Finally, the loss of attribute completion can be defined as:

$$L_a^{s/t} = \frac{1}{2\left(N^{s/t,o}\right)}\left\|X^{s/t,o} - \tilde{X}^{s/t,o}\right\|_2^2, \tag{8}$$

$$L_s^{s/t} = \frac{1}{\left(N^{s/t}\right)^2}\sum_{i=1}^{N^{s/t}}\sum_{j=1}^{N^{s/t}}\left\|\hat{A}_{ij}^{s/t} - \tilde{A}_{ij}^{s/t}\right\|_2^2, \tag{9}$$

$$L_d = L_a^{s/t} + L_s^{s/t}, \tag{10}$$

where $\tilde{A}^{s/t} = Z_s^{s/t}\left(Z^{s/t}\right)^T$, $L_a^{s/t}$ denotes the decoded attribute loss, $L_s^{s/t}$ denotes the decoded adjacency matrix loss, and L_d denotes total decoded loss.

3.4 Domain Adaptation Modeling

After obtaining the complementary source and target node embedding, we use the completed embedding as the input into domain classifier to capture domain-invariant of node embedding between the source and target graphs, making attribute complement and knowledge migration in a unified framework. Specifically, domain adaptation modeling consists of two components: a domain discriminator d and a node classifier c. The node classifier loss $L_c\left(Z^s, Y^s\right)$ can be defined as:

$$L_c\left(Z^s, Y^s\right) = -\frac{1}{N_s}\sum_{i=1}^{N_s} y_i^s log\left(\hat{y}_i^s\right), \tag{11}$$

where y_i^s denotes the label of node v_i in the source graph, $\hat{y}_i^s$ is the category prediction for node v_i in the source graph.

To mitigate the distribution discrepancy between graph domains caused by biases in attribute completion, we further provide supervision strategy for the completion of nodes in graphs. Inspired by [20], the loss of domain classifier L_{adv} is defined as:

$$\begin{aligned} L_{adv}\left(Z^s \otimes Y^s, Z^t \otimes \hat{Y}^t\right) = -\frac{1}{N_s+N_t}\sum_{i=1}^{N_s+N_t} & d_i \log\left(\hat{d}_i\right) \\ & + (1-d_i)\log\left(1-\hat{d}_i\right), \end{aligned} \tag{12}$$

where $d_i \in \{0, 1\}$ and $\hat{d}_i$ denotes the ground truth and domain prediction of node v_i in the source graph and target graph, respectively.

Finally, by integrating the above loss, the total loss of AC-GDA can be defined as:

$$L = L_c + L_d + L_{adv}. \tag{13}$$

4 Experiment

4.1 Experimental Setup

Datasets. We adopt citation networks from ArnetMiner [21] with the 0.82 homophily rate, consisting of Citationv1 (C), ACMv9 (A) and DBLPv7 (D), gamer networks from Twitch [22] with 0.55

Table 1: Classification accuracy (%) of cross-graph tasks for Blog and Twitch datasets.

	Blog		Twitch		
Methods	B1→B2	B2→B1	DE→ES	EN→RU	DE→PT
GCN	37.63±0.15	37.52±0.26	54.46±0.89	55.52±0.83	52.53±0.65
GAT	35.36±0.21	34.37±0.17	52.42±0.72	53.06±0.68	54.95±0.68
SAT	35.54±0.19	36.88±0.24	57.84±0.62	58.80±0.72	48.04±0.88
ITR	37.57±0.47	37.26±0.27	57.98±0.68	53.04±0.71	55.18±0.90
ASD-VAE	39.32±0.28	39.10±0.34	51.09±0.75	56.98±0.84	50.72±0.68
UDA-GCN	27.35±0.37	29.98±0.42	54.47±0.66	52.51±0.80	53.89±0.65
JHGDA	20.96±0.72	21.96±2.65	63.61±2.57	68.70±0.72	61.36±1.57
SGDA	19.83±0.20	27.04±0.42	53.32±0.38	55.27±0.40	54.13±0.48
PairAlign	26.68±0.36	28.63±0.57	52.46±0.36	52.58±0.36	49.90±0.53
StruRW	19.83±0.20	17.58±0.44	56.53±0.35	47.42±0.31	54.64±0.62
SpecReg	23.93±0.42	23.71±0.50	52.15±0.45	50.29±0.74	47.63±0.50
A2GNN	26.46±0.36	27.46±0.42	56.64±0.46	56.74±0.75	53.28±0.46
TDSS	22.13±0.32	24.42±0.41	54.81±0.24	60.53±0.24	54.28±0.32
AC-GDA	**42.25±0.61**	**40.50±0.09**	**64.87±0.68**	**68.66±0.88**	**62.63±0.94**

homophily rate, consisting of Germany (DE), England (EN), Spain (ES), France (FR), Portugal (PT) and Russia (RU), and social networks from Blog [22] with 0.40 homophily rate, consisting of Blog1 (B1) and Blog2 (B2).

Baselines. We compare our proposed AC-GDA with the following three categories of baselines:

- Classical graph convolutional networks methods: GCN [1] and GAT [23], these classical methods are trained on source graph and tested directly on target graph for unsupervised cross-domain tasks.
- Data imputation for attribute-missing graph methods: SAT [24], ITR [11], and ASD-VAE [25] these classical imputation methods are supervised by labels, which are also trained on source graph and tested directly on target graph for unsupervised tasks.
- Graph domain adaptation methods: A2GNN [6], SGDA [26], UDA-GCN [27], JHGDA [28], PairAlign [29], StruRW [30], SpecReg [31], and TDSS [32], these GDA methods pay attention to transferring label knowledge from source graph to target graph for unsupervised cross-graph tasks. However, these GDA methods only are applicable to complete graphs, unable to handle attribute-missing cross-domain tasks.

Implementation Details. The experiments are conducted in an environment with Python 3.8, PyTorch 2.0.0, and CUDA 11.8. We use all labeled source nodes and all unlabeled target nodes, and set the hidden layers from 512 to 64 in both source and target graphs. For our proposed AC-GDA, the dropout for each GCN layer is set to 0.3, and learning rate to $3e^{-3}$. In addition, we set attribute complement balance coefficient λ as 0.8, completion confidence threshold α as 0.4, and neighbor number k in completion confidence as 30, respectively. We follow the mask strategy proposed in ITR [11] to ensure a fair and consistent evaluation, that is, we randomly sample 40% nodes with attributes as the training set, and mask the rest attributes of 10% and 50% nodes (i.e., attribute-missing nodes) as the validation set and the test set in source and target graphs, respectively. For data imputation methods, we follow the default parameter settings in their original papers. For classical graph convolutional networks methods and graph domain adaptation methods, we mask the same ratio of attributes as our approach, and complete missing attributes with intra-graph 1-hop neighborhoods. For fair comparison, we run our proposed AC-GDA and all baselines ten times, and report the average value of ten best accuracy with standard deviation. Our source code and datasets are available at https://github.com/yangniya1/ACGDA.

4.2 Cross-graph Node Classification

Table 1 and Table 2 list the accuracy of different methods on cross-graph task with missing attributes. As shown, we can find that our proposed AC-GDA achieves the best performance across almost

Table 2: Classification accuracy (%) of cross-graph tasks for Citation dataset.

	ArnetMiner					
Methods	A→C	A→D	C→A	C→D	D→A	D→C
GCN	55.42± 0.42	50.12± 0.36	53.90± 0.61	53.95± 0.28	53.73± 0.42	51.45± 0.38
GAT	43.25± 0.26	39.33± 0.25	43.36± 0.25	43.45± 0.35	41.43± 0.24	40.46± 0.36
SAT	58.72± 0.35	54.08± 0.44	53.12± 0.32	53.22± 0.52	50.30± 0.58	50.90± 0.26
ITR	59.80± 0.36	54.07± 0.48	52.24± 0.47	55.41± 0.35	48.74± 0.36	54.08± 0.26
ASD-VAE	57.03± 0.50	58.46± 0.38	55.84± 0.51	58.74± 0.36	45.76± 0.38	53.18± 0.26
UDA-GCN	57.20±0.42	55.88±0.35	53.36±0.42	60.14±0.12	45.42±0.53	51.56±0.35
JHGDA	62.84±0.41	58.40±0.30	57.94±0.50	61.06±1.02	52.71±0.37	58.13±0.52
SGDA	56.36±0.73	56.94±0.44	55.82±0.28	57.04±0.42	50.52±0.38	57.42±0.48
PairAlign	48.70±0.71	54.23±0.30	48.42±0.49	57.89±1.48	47.93±0.47	42.75±0.81
StruRW	51.83±0.23	53.94±0.43	46.53±0.45	54.91±0.64	44.45±0.43	48.75±0.61
SpecReg	55.35±0.21	56.65±0.26	54.15±0.46	57.92±0.52	52.19±0.18	54.57±0.35
A2GNN	63.46±0.85	58.63±1.21	57.53±0.92	59.32±1.23	54.35±0.92	57.32±1.27
TDSS	64.47±1.37	60.02±0.86	58.46±0.87	60.30±0.83	55.33±1.35	58.03±0.84
AC-GDA	**65.72±0.22**	**61.90±0.14**	**59.10±0.21**	**61.04±0.43**	**55.29±0.39**	**57.96±0.79**

datasets with different homophily rate, which verifies the effectiveness of our main idea of jointly completing and dynamically optimizing attribute-missing nodes in an iterative manner. Notably, AC-GDA not only excels in high homophilic ArnetMiner network but also demonstrates outstanding performance in both general homophilic Twitch network and heterophilic Blog network, proving its strong generalization capability. The specific analysis is as follows:

Compared to classical graph convolutional networks methods, AC-GDA achieves the significant improvement for knowledge transferring on attribute-missing graphs. This is mainly because classical GCNs suffer from an inability to transfer abundant label knowledge from source graph to target graph, and unable to effectively learn node embedding of attribute-missing graphs, which demonstrates the effectiveness and superiority of our AC-GDA in unsupervised node classification task over attribute-missing graph. Similarly, AC-GDA outperforms existing data imputation methods for attribute-missing graphs. Such methods primarily focus on attribute completion within the graph by label supervision, and are not well-suited for unlabeled attribute-missing scenarios. This indicates both the knowledge completion and transferring between the source and target graphs are essential, highlighting the necessity and urgency of our proposed AC-GDA. More importantly, our proposed AC-GDA also outperforms the state-of-the-art GDA methods, which validates that leveraging cross-graph attribute and topology information to complete attribute-missing nodes is effective, and jointly completing and dynamically optimizing attribute-missing nodes in an iterative manner can promote positive knowledge transfer while completing attributes.

4.3 Ablation Study

Considering that AC-GDA contains multiple key components, we compare it with five variants:

- AC-GDA¬i: A variant of AC-GDA with the cross-graph attribute complement being removed, and only using topology embedding within the graph.
- AC-GDA¬a: A variant of AC-GDA with the completion confidence being removed.
- AC-GDA¬l: A variant of AC-GDA with removing the pseudo-label optimization and replacing it with consistency regularization.
- AC-GDA¬c: A variant of AC-GDA with the class domain alignment being removed, and only using domain alignment to capture domain invariants.
- AC-GDA¬cg: A variant of AC-GDA with intra-graph attributes, not cross-graph attributes for completion.

As shown in Figure 4, 1) Compared to AC-GDA¬i, the superior performance of AC-GDA shows topology embedding the graph is inadequate for attributes completion, illustrating the effectiveness of fully transferring cross-graph observed attributes to complete attribute-missing nodes. 2) Compared to AC-GDA¬a, the outstanding results of AC-GDA demonstrates that evaluating the confidence

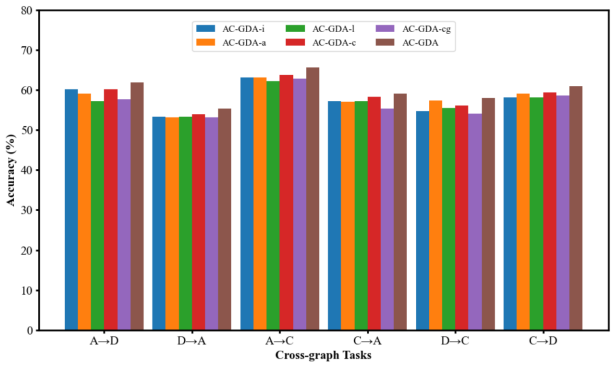
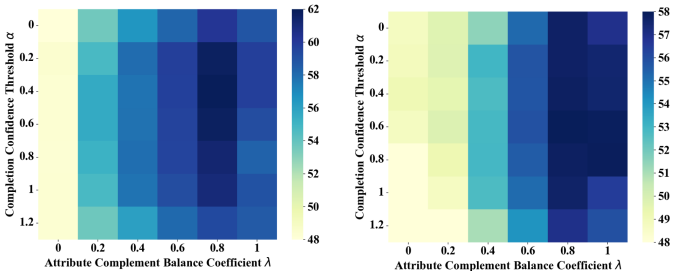


Figure 4: Classification accuracy (%) between AC-GDA variants on six cross-graph tasks.



(a) A→D (b) D→C

Figure 5: Analysis results of (a) the completion confidence threshold α, and (b) cross-graph attribute complement balance coefficient λ.

of complementary attributes helps to leverage the positive attribute information in the completion process, mitigating the phenomenon where excessive attribute-missing nodes and insufficient nodes are used for cross-domain attribute completion. 3) Compared to AC-GDA¬l, the notable improvement of AC-GDA implies optimizing pseudo-labels can mitigate falsely pseudo-labels for attribute-missing nodes, contributing to obtain more accurate completion confidence that guides trustworthy embedding can be dynamically expanded. 4) Compared to AC-GDA¬c, the superior performance of AC-GDA indicates that integrates cross-graph trustworthy embedding imputation and knowledge migration in a unified manner, therby enhanced node embeddings with label information can provide more effective supervision for the completion. 5) Compared with AC-GDA¬cg, our AC-GDA yields the improvement. This is because cross-graph completion not only enriches missing attributes, but also implicitly facilitates domain alignment.

4.4 Parameter Analysis

We further analyze the effect of missing ratios for our proposed AC-GDA, and estimate the sensitivity of three important parameters, that is, the cross-graph attribute complement balance coefficient λ, the completion confidence threshold α. Figure 5 shows the results of completion confidence threshold α by varying from 0 to 1.2 with an increment of 0.2, and cross-graph attribute complement balance coefficient λ by varying from 0 to 1 with an increment of 0.2 for cross-graph node classification.

Analysis of the Completion Confidence Threshold α. We can find that when α is less than 1, unsupervised node classification task captures the better performance. As α being too large may result that nodes with relatively low confidence are used for cross-graph attribute completion, while α being too small, insufficient nodes may overlook critical cross-graph attribute embedding for completion. This also indicates that the cross-graph attribute completion confidence is crucial for dynamically adjusting the scope of cross-graph attribute embeddings for completion, thereby improving the attribute-completion quality and the performance of unsupervised attribute-missing node classification.

Analysis of the Cross-graph Attribute Complement Balance Coefficient λ. We can find that when λ is set as 0.8 for the best classification results, the larger the balance coefficient λ, the better the

model performance, underling the importance of transferring cross-graph trustworthy attributes in our proposed AC-GDA. This also verifies that cross-graph trustworthy attributes is essential, but topology information within graph is also vital for attribute complement.

5 Related Work

Graph Domain Adaptation. Graph domain adaptation (GDA) technique achieve desirable performance on unsupervised cross-domain graph tasks [33, 34, 35] by transferring full label knowledge from source graph to unlabeled target graph. For instance, SPA [36] combines a coarse alignment mechanism with a spectral regularizer to align graph domains in feature space, and further develops a novel neighbor-aware self-training mechanism to enhance the discriminability of target domains. SDA [37] learns discriminative representations of target graphs by efficiently transferring knowledge to unlabeled target graph. AGLSP [38] learns personalized knowledge between nodes within classes and local knowledge of neighboring nodes from different classes to achieve multi-semantic granularity graph domain adaptation. GDABench [22], as the first comprehensive benchmark for unsupervised GDA, provides a standardized evaluation platform. After that, SAMGPT [39] designs a structure alignment method for text-free multi-domain graph pre-training and cross-domain adaptation. However, when faced with a cross-graph scenario where partial nodes' attributes are missing, existing GDA methods are not directly applicable due to the inability to learn effective representations. Thus, it is urgent to explore how to perform attribute-completion in cross-graph tasks.

Representation Learning for Attribute-Missing Graph. Researchers have developed many works to obtain node embedding for attribute-missing graph [40, 41, 42]. For example, Amer [15] completes missing attributes under the guidance of learning network representation via mutual information maximization, and imposes attribute-structure relationship constraint for completing missing attributes. AMGC [19] is proposed to train clustering and imputation together in a unified framework, iteratively generating cluster-enhanced nearest neighbor information for the data imputation. RAM-MVC [10] utilizes feature reconstruction to fill the missing attributes while constructing similarity graphs in feature spaces. TDAR [43] introduces an improved pre-filling method for initial attribute recovery using native graph topology, and dynamically adjusts the propagation weights. Unlike the above tasks on attribute-missing graph within a single domain, for the cross-graph task in attribute-missing graphs, the completely unlabeled target graph is unable to directly supervise the attribute complement. Therefore, we define attribute complement and knowledge migration into a unified framework to make full use of source labeled knowledge.

6 Conclusion

In this paper, we propose an adaptive attribute completion with representation space for incomplete graph domain adaptation (AC-GDA) to promote positive knowledge migration while completing attributes. We learn trustworthy embedding to complete attribute-missing nodes, and dynamically adjust the scope of trustworthy information, thereby reducing the distribution discrepancy between graphs. To prompt more sufficient knowledge transfer to target graph, we further enhance node embedding with labels to make nodes of the same category distribute consistently between graph domains. The experiment demonstrates the superiority of AC-GDA.

Border Impacts and Limitations. Our proposed AC-GDA fosters innovation and advances domain adaptation research on attribute-missing graph. However, benchmark datasets does not adequately represent the diversity and complexity of real-world data, such as multi-source or sequential adaptation, which can lead to unrealistic expectations. We plan to improve AC-GDA to adapt to complex and dynamic real-world scenarios, which is crucial for the development of graph domain adaption.

7 Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62276187, No. 62422210, No. 62402337, No. 92370111, No. 62272340, No. 92570118 and No. 623760888), the Hebei Natural Science Foundation (No. F2024202047), the Hebei Yanzhao Golden Platform Talent Gathering Programme Core Talent Project (Education Platform) (HJZD202509)

References

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

[2] Tianzi Zang, Yanmin Zhu, Haobing Liu, Ruohan Zhang, and Jiadi Yu. A survey on cross-domain recommendation: taxonomies, methods, and future directions. *ACM Transactions on Information Systems*, 41(2):1–39, 2022.

[3] Xiaolong Fan, Maoguo Gong, Yue Wu, Mingyang Zhang, Hao Li, and Xiangming Jiang. Ccgib: A cross-channel graph information bottleneck principle. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–12, 2024.

[4] Chen Xu, Qiang Wang, Wenqi Zhang, and Chen Sun. Spatiotemporal ego-graph domain adaptation for traffic prediction with data missing. *IEEE Transactions on Intelligent Transportation Systems*, 25(12):20804–20819, 2024.

[5] Jun Xiao, Lin Li, Dejing Xu, Chengjiang Long, Jian Shao, Shifeng Zhang, Shiliang Pu, and Yueting Zhuang. Explore video clip order with self-supervised and curriculum learning for video applications. *IEEE Transactions on Multimedia*, 23:3454–3466, 2020.

[6] 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*, pages 13963–13971, 2024.

[7] 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):1–24, 2024.

[8] Zhen Zhang, Meihan Liu, Anhui Wang, Hongyang Chen, Zhao Li, Jiajun Bu, and Bingsheng He. Collaborate to adapt: Source-free graph domain adaptation via bi-directional adaptation. In *Proceedings of the ACM on Web Conference 2024*, pages 664–675, 2024.

[9] Gabriele Corso, Hannes Stark, Stefanie Jegelka, Tommi Jaakkola, and Regina Barzilay. Graph neural networks. *Nature Reviews Methods Primers*, 4(1):17, 2024.

[10] Dayu Hu, Suyuan Liu, Jun Wang, Junpu Zhang, Siwei Wang, Xingchen Hu, Xinzhong Zhu, Chang Tang, and Xinwang Liu. Reliable attribute-missing multi-view clustering with instance-level and feature-level cooperative imputation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 1456–1466, 2024.

[11] Wenxuan Tu, Sihang Zhou, Xinwang Liu, Yue Liu, Zhiping Cai, En Zhu, Changwang Zhang, and Jieren Cheng. Initializing then refining: A simple graph attribute imputation network. In *Proceedings of the 31th International Conference on Artificial Intelligence*, pages 3494–3500, 2022.

[12] Hongchang Gao and Heng Huang. Deep attributed network embedding. In *Proceedings of the 27th International Conference on Artificial Intelligence*, pages 3364–3370, 2018.

[13] Patrick Schober, Christa Boer, and Lothar A Schwarte. Correlation coefficients: appropriate use and interpretation. *Anesthesia & analgesia*, 126(5):1763–1768, 2018.

[14] Guanghui Zhu, Zhennan Zhu, Wenjie Wang, Zhuoer Xu, Chunfeng Yuan, and Yihua Huang. Autoac: Towards automated attribute completion for heterogeneous graph neural network. In *2023 IEEE 39th International Conference on Data Engineering*, pages 2808–2821, 2023.

[15] Di Jin, Rui Wang, Tao Wang, Dongxiao He, Weiping Ding, Yuxiao Huang, Longbiao Wang, and Witold Pedrycz. Amer: A new attribute-missing network embedding approach. *IEEE Transactions on Cybernetics*, 53(7):4306–4319, 2022.

[16] Mengran Li, Yong Zhang, Wei Zhang, Shiyu Zhao, Xinglin Piao, and Baocai Yin. Csat: Contrastive sampling-aggregating transformer for community detection in attribute-missing networks. *IEEE Transactions on Computational Social Systems*, 11(2):2277–2290, 2024.

[17] Gianni Brauwers and Flavius Frasincar. A general survey on attention mechanisms in deep learning. *IEEE transactions on knowledge and data engineering*, 35(4):3279–3298, 2021.

[18] Xinyu Zhang, Meng Kang, and Shuai Lü. Low category uncertainty and high training potential instance learning for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 16881–16889, 2024.

[19] Wenxuan Tu, Renxiang Guan, Sihang Zhou, Chuan Ma, Xin Peng, Zhiping Cai, Zhe Liu, Jieren Cheng, and Xinwang Liu. Attribute-missing graph clustering network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 15392–15401, 2024.

[20] Xiao Shen, Shirui Pan, Kup-Sze Choi, and Xi Zhou. Domain-adaptive message passing graph neural network. *Neural Networks*, 164:439–454, 2023.

[21] Jundong Li, Xia Hu, Jiliang Tang, and Huan Liu. Unsupervised streaming feature selection in social media. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1041–1050, 2015.

[22] Meihan Liu, Zhen Zhang, Jiachen Tang, Jiajun Bu, Bingsheng He, and Sheng Zhou. Revisiting, benchmarking and understanding unsupervised graph domain adaptation. In *Proceedings of the Advances in Neural Information Processing Systems*, 2024.

[23] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *Proceedings of the International Conference on Learning Representations*, 2018.

[24] Xu Chen, Siheng Chen, Jiangchao Yao, Huangjie Zheng, Ya Zhang, and Ivor W Tsang. Learning on attribute-missing graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(2):740–757, 2020.

[25] Xinke Jiang, Zidi Qin, Jiarong Xu, and Xiang Ao. Incomplete graph learning via attribute-structure decoupled variational auto-encoder. In *Proceedings of the 17th ACM international conference on web search and data mining*, pages 304–312, 2024.

[26] Ziyue Qiao, Xiao Luo, Meng Xiao, Hao Dong, Yuanchun Zhou, and Hui Xiong. Semi-supervised domain adaptation in graph transfer learning. In *Proceedings of the 32th International Conference on Artificial Intelligence*, pages 2279–2287, 2023.

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

[28] Boshen Shi, Yongqing Wang, Fangda Guo, Jiangli Shao, Huawei Shen, and Xueqi Cheng. Improving graph domain adaptation with network hierarchy. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 2249–2258, 2023.

[29] 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*, pages 32552–32575, 2024.

[30] Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiu Qiang, and Pan Li. Structural re-weighting improves graph domain adaptation. In *Proceedings of the 40th International Conference on Machine Learning*, pages 21778–21793, 2023.

[31] Yuning You, Tianlong Chen, Zhangyang Wang, and Yang Shen. Graph domain adaptation via theory-grounded spectral regularization. In *The eleventh international conference on learning representations*, 2023.

[32] 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, pages 15875–15883, 2025.

[33] Siyang Luo, Ziyi Jiang, Zhenghan Chen, and Xiaoxuan Liang. Domain adaptive graph classification. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4915–4919, 2024.

[34] Nan Yin, Mengzhu Wang, Zhenghan Chen, Li Shen, Huan Xiong, Bin Gu, and Xiao Luo. Dream: Dual structured exploration with mixup for open-set graph domain adaption. In *Proceedings of the 20th International Conference on Learning Representations*, 2024.

[35] Minying Zhang, Kai Liu, Yidong Li, Shihui Guo, Hongtao Duan, Yimin Long, and Yi Jin. Unsupervised domain adaptation for person re-identification via heterogeneous graph alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3360–3368, 2021.

[36] Zhiqing Xiao, Haobo Wang, Ying Jin, Lei Feng, Gang Chen, Fei Huang, and Junbo Zhao. Spa: a graph spectral alignment perspective for domain adaptation. *Advances in Neural Information Processing Systems*, 36, 2024.

[37] Yu Wang, Ronghang Zhu, Pengsheng Ji, and Sheng Li. Open-set graph domain adaptation via separate domain alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9142–9150, 2024.

[38] Zefeng Zheng, Shaohua Teng, Luyao Teng, Wei Zhang, and NaiQi Wu. Adaptive graph learning with semantic promotability for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[39] Xingtong Yu, Zechuan Gong, Chang Zhou, Yuan Fang, and Hui Zhang. Samgpt: Text-free graph foundation model for multi-domain pre-training and cross-domain adaptation. In *Proceedings of the ACM on Web Conference 2025*, pages 1142–1153, 2025.

[40] Nikhil Chowdary Nettem, Aashish Gopal Nukala, Siruguppa Vishnu Sai Kaushal, et al. Inference of missing attributes in multilevel complex network. In *2024 5th International Conference on Electronics and Sustainable Communication Systems*, pages 675–682, 2024.

[41] Hejie Cui, Zijie Lu, Pan Li, and Carl Yang. On positional and structural node features for graph neural networks on non-attributed graphs. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*, pages 3898–3902, 2022.

[42] Yushun Dong, Ninghao Liu, Brian Jalaian, and Jundong Li. Edits: Modeling and mitigating data bias for graph neural networks. In *Proceedings of the ACM Web Conference 2022*, pages 1259–1269, 2022.

[43] Mengran Li, Junzhou Chen, Chenyun Yu, Guanying Jiang, Ronghui Zhang, Yanming Shen, and Houbing Herbert Song. Topology-driven attribute recovery for attribute missing graph learning in social internet of things. *IEEE Internet of Things Journal*, pages 1–1, 2025.

Appendix

A Detailed Description of Datasets

We provide detailed description of datasets, as shown in Table 3.

- **ArnetMiner** dataset is citation networks derived from academic publication records where nodes represent research papers, edges denote citation relationships between papers, and papers are divided into five research categories: databases, artificial intelligence, computer vision, information security, and networking. Three specific networks are commonly used: ACMv9, Citationv1, and DBLPv7. ACMv9 includes ACM publications from 2000 to 2010, Citationv1 contains papers extracted from the Microsoft Academic Graph before 2008, DBLPv7 gathers papers indexed by DBLP between 2004 and 2008. These datasets serve as classical benchmarks for evaluating unsupervised graph domain adaptation.
- **Blog** dataset commonly refers as BlogCatalog, which is a real-world graph-constructed social network from an online blogging platform. In this dataset, nodes correspond to bloggers, and edges indicate social relationships such as follower or friend links between bloggers. Each blogger is associated with a high-dimensional feature vector, typically derived from profile metadata or textual content. Each blogger is assigned to the interest category (e.g., technology, sports, education, health, business, and entertainment). BlogCatalog generally is divided into Blog1 and Blog2 to evaluate unsupervised graph domain adaptation.
- **Twitch** dataset is a social network from the Twitch live-streaming platform, where nodes represent Twitch users, edges capture social interactions such as follows or friendships between Twitch users. Each user is associated with a feature vector that describe profile-level metadata, activity information, and language-specific attributes. In addition, each user is divided into two categories based on their language region, the Twitch dataset also supports node classification or graph domain adaptation.

Table 3: Statistics of datasets.

Datasets	#Nodes	#Edges	#Attributes	#Labels
Citationv1 (C)	8,935	15,113	6775	5
ACMv9 (A)	9,360	15,602		
DBLPv7 (D)	5,484	8,130		
Blog1 (B1)	2,300	66,942	8,189	6
Blog2 (B2)	2,895	107,672		
England (EN)	7,126	35,324	3,170	2
Germany (GE)	9,498	153,138		
France (FR)	6,549	112,666		
Russia (RU)	4,385	37,304		
Spain (ES)	4,648	59,382		
Porutgal (PT)	1,912	31,299		

B Detailed Description of Baselines

We provide detailed description of baselines.

- **GCN** aggregates features from neighboring nodes to learn node embeddings. The code is available at https://github.com/tkipf/pygcn.
- **GAT** leverages self-attention mechanisms to learn the importance of neighboring nodes, enabling adaptive aggregation of information from neighboring nodes. The code is available at https://github.com/PetarV-/GAT.
- **SAT** makes a shared-latent space assumption on graphs and develops a novel distribution matching-based GNN called structure-attribute transformer for attribute-missing graphs. The code is available at https://github.com/xuChenSJTU/SAT-master-online.

- **ITR** employs the structure information for initial imputation, and then leverages observed attribute and structure information to adaptively refine the imputed latent variables. The code is available at https://github.com/WxTu/ITR.
- **ASD-VAE** separately encodes attributes and structures, generating representations for each view, the missing values of attributes are imputed from this learned latent space. The code is available at https://github.com/jiangxinke/ASD-VAE.git.
- **UDA-GCN** is classical unsupervised domain adaptive graph convolutional networks for graph domain adaptation learning. The code is available at https://github.com/GRAND-Lab/UDAGCN.
- **JHGDA** focuses on each hierarchical level to characterize distinct and meaningful properties of the original graph, integrating domain discrepancy based on such hierarchies to a more precise domain discrepancy measurement. The code is available at https://github.com/Skyorca/JHGDA.
- **SGDA** adds adaptive shift parameters to source and target nodes, and measures the posterior influence of nodes based on their relative position with the class centroids to address the label scarcity. The code is available at https://github.com/joe817/SGDA.
- **PairAlign** uses edge weights to recalibrate the influence among neighboring nodes to mitigate conditional structure shift, meanwhile adjusting the classification loss with label weights to handle label shift. The code is available at https://github.com/Graph-COM/Pair-Align.
- **StruRW** solves different impacts of distribution shifts caused by either graph structure or node attributes and defines the conditional structure shift. The code is available at https://github.com/Graph-COM/StruRW.
- **SpecReg** designs theory-grounded algorithms for graph domain adaptation, and extends the GDA theory into the conditional shift scenario. The code is available at https://github.com/Shen-Lab/GDA-SpecReg.
- **A2GNN** re-evaluates the role of GNNs in graph domain adaptation, and uncovers the pivotal role of the propagation process for adapting to different graph domains. The code is available at https://github.com/Meihan-Liu/24AAAI-A2GNN.
- **TDSS** performs structural smoothing directly on the target graph, thereby mitigating structural distribution shifts and ensuring the consistency of node representations. The code is available at https://github.com/cwei01/TDSS.

C Analysis of Missing Rates

As shown in Table 4, when the missing ratio increases from 0% to 80%, AC-GDA shows a much slower performance drop compared with UDA-GCN on ArnetMiner, even steady performance on Blog and Twitch as the missing rate increases. It is because that AC-GDA complete attributes with cross-graph trustworthy attribute embeddings, promoting positive knowledge migration while completing attributes. From dataset perspective, A, C and D in ArnetMiner originate from different years, exhibiting relatively specific distributions. As the missing rate increases, cross-domain attribute completion introduces some irrelevant information, leading to a certain decline in performance. In contrast, in the Twitch and Blog datasets, graph domains derive from network relationships occurring at the same time, resulting in relatively smaller specific distributions. As the missing rate increases, cross-domain attribute completion remains more effective, thereby maintaining stable performance. It also demonstrates the effectiveness and superiority of our proposed cross-domain attribute completion. Furthermore, we designed the dynamic trustworthy embedding adjustment strategy able to divert trusted complementary nodes V_m to the set of attribute-observed nodes V_e while completing attributes, avoiding semantic deviation caused by insufficient available attributes for completion when the number of missing attributes is too high.

To clarify that cross-graph attribute completion is consistently superior to in-graph attribute completion, we further report the absolute gain and uncertainty comparison, as shown in Table 5. Compared to AC-GDA¬cg, AC-GDA yields the improvement and cross-graph gain is statistically significant. This is because cross-graph completion not only enriches missing attributes, but also implicitly facilitates domain alignment.

Table 4: Results of cross-graph task under different missing ratios.

Tasks	Methods	0	20%	40%	60%	80%
A→C	UDA-GCN	71.80	68.56	64.17	57.20	52.25
	AC-GDA	76.07	74.82	70.89	65.72	60.25
C→A	UDA-GCN	70.18	63.68	54.38	53.36	46.05
	AC-GDA	72.31	67.05	62.46	59.10	52.64
D→C	UDA-GCN	72.15	62.09	53.62	51.56	44.64
	AC-GDA	74.26	68.04	63.73	57.96	54.46
B1→B2	UDA-GCN	27.18	25.04	22.81	27.35	23.29
	AC-GDA	42.06	42.92	45.03	42.25	30.92
B2→B1	UDA-GCN	25.28	28.04	24.83	29.98	25.82
	AC-GDA	42.25	38.95	40.35	40.50	32.92
DE→ES	UDA-GCN	56.72	53.52	47.64	54.47	48.17
	AC-GDA	60.47	64.37	63.27	64.87	64.74
EN→RU	UDA-GCN	53.48	48.83	54.54	52.51	54.46
	AC-GDA	60.56	64.36	67.26	68.66	67.64
DE→PT	UDA-GCN	54.27	49.53	54.06	53.89	46.53
	AC-GDA	61.67	62.48	58.67	62.63	60.63

Table 5: Ablation results of AC-GDA and AC-GDA¬cg on cross-domain tasks.

Tasks	AC-GDA (%)	AC-GDA-¬cg (%)	Absolute Gain (%)	Statistical Significance
A→D	61.90 ± 0.14	57.74 ± 0.21	4.16	$p < 0.001$
D→A	55.29 ± 0.39	53.16 ± 0.30	2.13	$p < 0.001$
A→C	65.72 ± 0.22	62.80 ± 0.37	2.92	$p < 0.001$
C→A	59.10 ± 0.21	55.42 ± 0.18	3.68	$p < 0.001$
D→C	57.96 ± 0.79	54.16 ± 0.54	3.80	$p < 0.001$
C→D	61.04 ± 0.43	58.63 ± 0.36	2.41	$p < 0.001$

We also provide the visual analysis with different missing ratios. Figures 6 to 8 illustrate the performance on cross-graph node classification under varying missing ratios, where the value of missing ratios is relatively stable between 0 and 80%, indicating that compared to four SOTA baselines, the AC-GDA performance diminishes not particularly strongly as the missing rate increases. This also verifies that cross-graph trustworthy attributes and topology information within the graph can effectively complete the attribute-missing nodes to prevent the sharp decrease of AC-GDA with the increase of missing rate, and achieving the label knowledge migration of the source graph to the target graph, improving the prediction performance of the target graph.

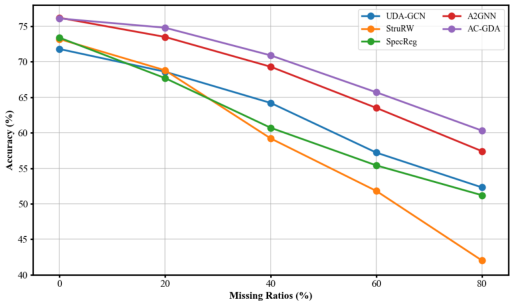


Figure 6: Performance comparison between AC-GDA and SOTA baselines with different missing ratios on A→C tasks.

D Analysis of the Neighbors Number k in the Completion Confidence

We first retrieve the k-nearest neighbors based on Euclidean distances, denoted as $\mathcal{N}_i$, and then average the neighbor prediction probabilities p_j^t from shared classifier, thereby formulating the

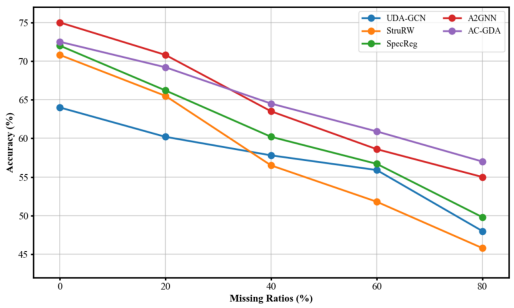


Figure 7: Performance comparison between AC-GDA and SOTA baselines with different missing ratios on A→D tasks.

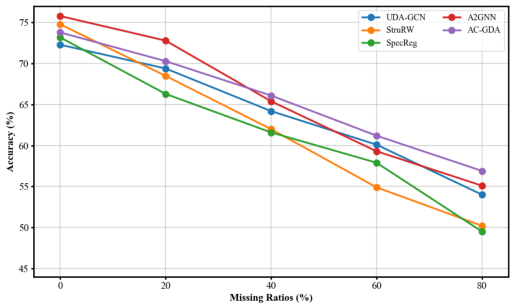


Figure 8: Performance comparison between AC-GDA and SOTA baselines with different missing ratios on C→D tasks.

pseudo prediction probability $\hat{q}_i^t$ for target node v_i^t:

$$\hat{q}_i^t = \mathbb{E}_{x_j^t \in \mathcal{N}_i} p_j^t. \tag{14}$$

Finally, we derive the pseudo-label $\hat{y}_i^t$:

$$\hat{y}_i^t = \arg\max_m \hat{q}_{i,m}^t. \tag{15}$$

Figure 9 shows the results of neighbors number k in the completion confidence for cross-graph node classification, where neighbors number k varies between 10 and 60 for the classification results. This emphasizes that it is crucial to dynamically adjust the range of trustworthy attributes across graph, as k being too small may result in ignoring related neighbors.

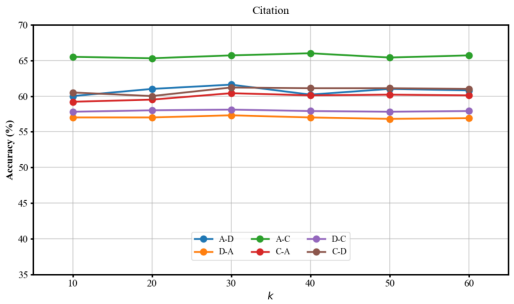
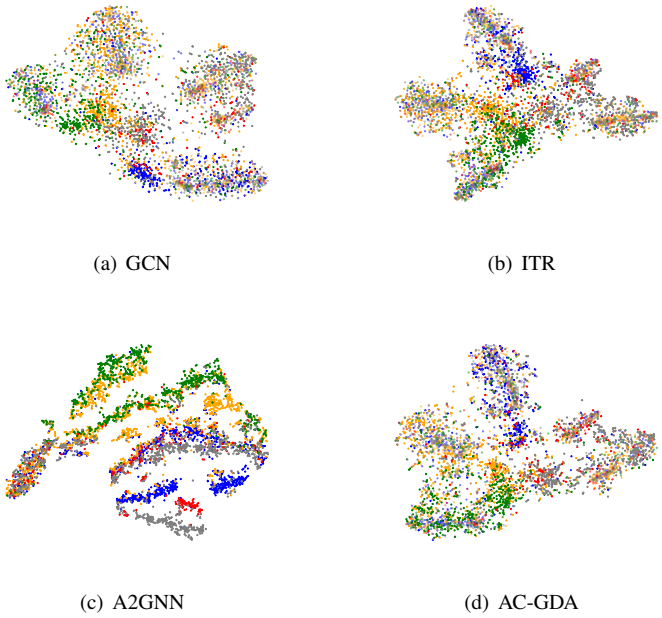


Figure 9: Performance of AC-GDA with different neighbor number k.

E Visualization

To provide a more intuitive comparison, we take the A→C task as an illustrative example, and visualize our AC-GDA and three baselines by t-SNE in a two-dimensional space.

As shown in Figure 10, AC-GDA exhibits clearer boundaries compared to other three baselines. That is, nodes belonging to the same category are closely clustered, while nodes from different categories



(a) GCN (b) ITR

(c) A2GNN (d) AC-GDA

Figure 10: Visualization of target node embeddings on task of A→C via t-SNE, where each color indicates a category.

are separated. This demonstrates that our AC-GDA can effectively distinguish different classes in the embedding space. In addition, nodes with the same label from both source and target domains are grouped together and have the overlapping clusters, suggesting that AC-GDA effectively reduces domain discrepancy. Notably, such separation is indicative of great performance on unsupervised node classification task.

F Discussion of Topology Structure Noise

Classical attribute completion methods (i.e., zero-filling, parameter-filling, and graph-aware filling) that may introduce a significant amount of irrelevant information, due to target graph is completely unlabeled and unsupervised, which may exacerbate the distribution discrepancy. We propose a joint completion mechanism guided by cross-graph topology similarity, which first initializes the missing attributes using topology-guided similarity in the representation space, viewing cross-graph topology similarity as completion weight and the corresponding attribute embedding to fill. Then we refine and constrain node embeddings through completion confidence and neighbor supervision to provide supervision for completed nodes, and make nodes of the same category distribute consistently across graph domains. This joint optimization makes attribute completion and the learning of cross-graph trustworthy embeddings mutually reinforcing in an iterative manner, improving the robustness and effectiveness of this similarity guidance. In this way, AC-GDA can mitigate the interference caused by topological noise on attribute completion.

G Analysis of the Time/Space Complexity

Supposing there are source graph with Ns nodes and Ms edges and target graph with N_t nodes and M_t edges, the node representation dimension, layer and category are set as D, L and K. The time complexity of node embedding, attribute completion and domain adaptation are $O(LN_sD^2+LM_sD+LN_tD^2+LM_tD)$, $O(N_sD^2+N_tD^2+DN_s\log(N_s)+DN_t\log(N_t))$ and $O(KN_sD^2+KN_tD^2)$, respectively. In our experiment, performing principal component analysis and reducing dimension D to 1000, hence the asymptotic complexity is bounded by $O(N_s\log(N_s)+M_s+N_t\log(N_t)+M_t)$.

Meanwhile, to store the node attribute embedding, topology embedding and category prediction, AC-GDA requires a space complexity of $O(N_sD + M_sD + N_sK + N_tD + M_tD + N_tK)$.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See Abstract and Introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Conclusion.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See Section 3, we propose the Attribute-Topology Similarity Consistency hypothesis. To verify whether our assumption holds, we randomly select 1000 node pairs on Citation dataset with 60% missing ratio, compute their topology similarity and attribute similarity, respectively, and give qualitative analysis and quantitative analysis.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines: See Section 4.

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: See Github.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: See Section 4.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [N/A]

Justification: Our experiments are only for directly validating the proposed attribute completion mechanism rather than competing with state-of-the-art methods, and we prove our proposed adaptive completion mechanism always works effectively with different missing rate, as shown in Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All authors make sure to preserve anonymity.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Section 6.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The paper poses no such risk.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The used assets (e.g., datasets, codes, models) are properly credited in the submissions. We have carefully followed the corresponding licenses and terms of use

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Our newly introduced code and datasets are included in the github or provided in Appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: This submission does not involve LLMs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.