Revisiting Initializing Then Refining: An Incomplete and Missing Graph Imputation Network

Wenxuan Tu, Bin Xiao, Xinwang Liu, *Senior Member, IEEE*, Sihang Zhou, Zhiping Cai, *Member, IEEE*, and Jieren Cheng

***Abstract*— With the development of various applications, such as recommendation systems and social network analysis, graph data have been ubiquitous in the real world. However, graphs usually suffer from being absent during data collection due to copyright restrictions or privacy-protecting policies. The graph absence could be roughly grouped into attribute-incomplete and attribute-missing cases. Specifically, attribute-incomplete indicates that a portion of the attribute vectors of all nodes are incomplete, while attribute-missing indicates that all attribute vectors of partial nodes are missing. Although various graph imputation methods have been proposed, none of them is custom-designed for a common situation where both types of graph absence exist simultaneously. To fill this gap, we develop a novel graph imputation network termed revisiting initializing then refining (RITR), where both attribute-incomplete and attribute-missing samples are completed under the guidance of a novel initializing-then-refining imputation criterion. Specifically, to complete attribute-incomplete samples, we first initialize the incomplete attributes using Gaussian noise before network learning, and then introduce a structure-attribute consistency constraint to refine incomplete values by approximating a structure-attribute correlation matrix to a high-order structure matrix. To complete attribute-missing samples, we first adopt structure embeddings of attribute-missing samples as the embedding initialization, and then refine these initial values by adaptively aggregating the reliable information of attribute-incomplete samples according to a dynamic affinity structure. To the best of our knowledge, this newly designed method is the first end-to-end unsupervised framework dedicated to handling hybrid-absent graphs. Extensive experiments on six datasets have verified that our methods consistently outperform the existing state-of-the-art competitors. Our source code is available at https://github.com/WxTu/RITR.**

***Index Terms*— Feature completion, graph neural network (GNN), hybrid-absent data, incomplete multiview learning.**

Manuscript received 15 February 2023; revised 16 October 2023; accepted 22 December 2023. Date of publication 12 January 2024; date of current version 6 February 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62325604, Grant 62276271, Grant 62006237, Grant 62162024, and Grant 62162022; in part by the Hunan Provincial Natural Science Foundation of China under Grant 2021JJ30779; in part by the Science and Technology Innovation Program of Hunan Province under Grant 2022RC3061; and in part by the Postgraduate Scientific Research Innovation Project in Hunan Province under Grant CX20220076. *(Corresponding authors: Sihang Zhou; Xinwang Liu.)*

Wenxuan Tu, Xinwang Liu, and Zhiping Cai are with the School of Computer, National University of Defense Technology, Changsha 410073, China (e-mail: wenxuantu@163.com; xinwangliu@nudt.edu.cn; zpcai@nudt.edu.cn).

Bin Xiao is with the Department of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: xiaobin@cqupt.edu.cn).

Sihang Zhou is with the School of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China (e-mail: sihangjoe@gmail.com).

Jieren Cheng is with the School of Computer Science and Technology, Hainan University, Haikou 570228, China, and also with the Hainan Blockchain Technology Engineering Research Center, Haikou 570228, China (e-mail: cjr22@163.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2024.3349850, provided by the authors.

Digital Object Identifier 10.1109/TNNLS.2024.3349850

NOMENCLATURE

$\mathbf{X} \in \mathbb{R}^{N\times D}$	Original attribute matrix.
$\mathbf{X}^I \in \mathbb{R}^{N^I\times D}$	Incomplete attribute matrix.
$\mathbf{N} \in \mathbb{R}^{N^I\times D}$	Gaussian noise matrix.
$\widetilde{\mathbf{X}}^I \in \mathbb{R}^{N^I\times D}$	Corrupted incomplete attribute matrix.
$\mathbf{A} \in \mathbb{R}^{N\times N}$	Original adjacency matrix.
$\widetilde{\mathbf{A}} \in \mathbb{R}^{N\times N}$	Normalized adjacency matrix.
$\widetilde{\mathbf{A}}^I \in \mathbb{R}^{N^I\times N^I}$	Normalized adjacency matrix of $\mathcal{G}^{\text{Sub}}$.
$\mathbf{I} \in \mathbb{R}^{N\times N}$	Identity matrix.
$\mathbf{C} \in \mathbb{R}^{N^I\times N^I}$	Structure-attribute correlation matrix.
$\widetilde{\mathbf{A}}^{I(o)} \in \mathbb{R}^{N^I\times N^I}$	o-order normalized adjacency matrix of $\mathcal{G}^{\text{Sub}}$.
$\mathbf{H}_A^I \in \mathbb{R}^{N^I\times d}$	Attribute embeddings of incomplete samples.
$\mathbf{H}_S \in \mathbb{R}^{N\times d}$	Structure embeddings.
$\mathbf{H}_S^M \in \mathbb{R}^{N^M\times d}$	Structure embeddings of missing samples.
$\mathbf{H}_I \in \mathbb{R}^{N\times d}$	Initially imputed embeddings.
$\mathbf{R} \in \mathbb{R}^{N\times N}$	Affinity structure.
$\mathbf{H} \in \mathbb{R}^{N\times d}$	Imputed embeddings.
$\mathbf{H}^M \in \mathbb{R}^{N^M\times d}$	Imputed embeddings of missing samples.
$\widetilde{\mathbf{H}} \in \mathbb{R}^{N\times d}$	Sample-recomposed embeddings.
$\mathbf{S} \in \mathbb{R}^{N\times N}$	Normalized self-correlated matrix.
$\mathbf{M} \in \mathbb{R}^{N^I\times D}$	Elementwise indicator matrix.
$\widehat{\mathbf{X}} \in \mathbb{R}^{N\times D}$	Rebuilt attribute matrix.
$\widehat{\mathbf{X}}^I \in \mathbb{R}^{N^I\times D}$	Rebuilt attribute matrix of incomplete samples.
$\widehat{\mathbf{A}} \in \mathbb{R}^{N\times N}$	Rebuilt adjacency matrix.

I. INTRODUCTION

GRAPHS, which model and represent the complicated relationships among real-world objects, are ubiquitous in practical scenarios, including citation graphs, social graphs, protein graphs, and molecule graphs. To analyze the graph data, graph machine learning attempts to transform an original graph into low-dimensional representations by preserving node

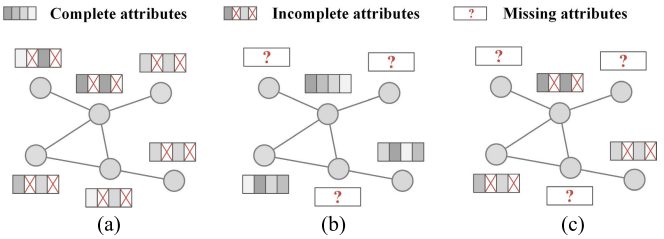


Fig. 1. Different types of absent graphs. (a) Attribute-incomplete graph: particular attributes of all the samples are absent. (b) Attribute-missing graph: all the attributes of specific nodes are absent. (c) Hybrid-absent graph: both circumstances (a) and (b) exist simultaneously within a common graph. As one can easily see, the last category is the most challenging. However, it is still underexplored in previous literature. We make the first attempt to solve it by proposing a novel method called RITR.

attributes and graph structure simultaneously [1], [2], [3]. In recent years, with the help of graph neural networks (GNNs), graph machine learning has become an increasingly powerful artificial intelligence technique. It has achieved significant success in diverse real-world applications, such as anomalous citation detection [4], few-shot learning [5], and knowledge graphs [6], [7].

The key prerequisite for the impressive performance of the existing graph machine learning methods lies in the assumption that all the samples within a graph are available and complete. However, this assumption may not always hold in practice since it is hard to collect all the information from graph data. The reasons behind this include but are not limited to privacy-protecting policies, copyright restrictions, and simply not enough information. For example, in a co-purchase graph, consumers tend to selectively (or entirely not) provide their feedback for specific items due to privacy concerns. In a citation graph, some papers are inaccessible due to copyright protection. All these circumstances could easily trigger sparsity and data-absent problems that adversely affect the learned representations. According to the type of node attribute absence, the absent graphs can be roughly divided into two categories: 1) the attribute-incomplete graph where only a portion of attributes of all the nodes are absent and 2) the attribute-missing graph where all the attributes of specific nodes are absent. Fig. 1 illustrates the situations of attribute absence. Among them, Fig. 1(a) corresponds to an attribute-incomplete graph, Fig. 1(b) corresponds to an attribute-missing graph, and Fig. 1(c) corresponds to a hybrid-absent graph where both the attribute-incomplete samples and attribute-missing samples exist in the same graph. The above cases make valuable information invisible and pose significant challenges to the existing graph machine learning methods for graph analysis.

To tackle the attribute-incomplete learning problem, many efforts have been devoted to developing various imputation strategies such as matrix completion [8], [9], generative adversarial network (GAN) [10], Gaussian mixture model (GMM) [11], and other advanced ones [12], [13]. To impute incomplete attributes, these methods integrate a standard GNN-based framework and data imputation techniques to conduct the sample embedding. Although significant progress has been made in solving the attribute-incomplete learning problem, the performance of these methods degrades drastically when they encounter extremely absent data (e.g., attribute-missing graphs). To solve this issue, a recent advanced method termed SAT [14] first introduces an unsupervised graph imputation framework to handle attribute-missing graphs under the guidance of a shared-latent space assumption. Although achieving encouraging success, SAT suffers from the following limitations when conducting the data imputation.

1) *Two-Source Information Isolation:* SAT isolates the learning processes of embeddings of observed node attributes and the complete graph structure. This prevents the trustworthy visible information from being sufficiently used, which could cause the learned representations to be biased and increase the risk of inaccurate data imputation.
2) *Strict Prior Assumption:* SAT forces two-source latent variables to align with an in-discriminative noise matrix obeying normal distribution. While in reality, the predefined normal distribution would not ideally conform to the complex graphs. As a result, the negotiation between attribute and structure information tends to get overly rigid, resulting in less discriminative representations. This could adversely affect the quality of the rebuilt attribute matrix of all the samples, especially those without attributes.

To forbid the adverse effect of inaccurate simple initialization and the limitation of rigid distribution assumption, we propose an initializing then refining (ITR) method [15]. While there is potential for effectively tackling the attribute-missing problem, we have observed that when ITR processes hybrid-absent graphs, attribute-incomplete samples would largely undermine the quality of generated attribute-missing features. This is primarily due to the diffusion of inaccurate imputed information. To our knowledge, hybrid-absent graph machine learning has not been studied in the existing graph literature, which is a universal yet challenging problem in various practical applications. To fill this gap, we revisit ITR and further improve it by designing a variant, termed revisiting ITR (RITR). In this newly proposed RITR, we complete both the attribute-incomplete and attribute-missing samples under the guidance of the initializing-then-refining imputation criterion. To impute incomplete attributes, we elaborately design a sample-denoising then consistency-preserving (STC) mechanism. As illustrated in Fig. 2, the feature completion process within this mechanism mainly includes step 1 to step 3. First, we use a denoising learning approach that combines the attribute and structure information of attribute-incomplete samples to learn sample embeddings. Second, we consider all the nodes in the graph and focus on learning sample embeddings solely based on the structure information. Finally, we introduce a structure-attribute consistency constraint to refine these incomplete latent variables by approximating a structure-attribute correlation matrix to a high-order structure matrix. This operation aims to ensure the representation quality of nodes with incomplete attributes, so as to provide a feasible initialization for nodes with missing attributes in the subsequent steps. To impute missing attributes, we design another data imputation mechanism termed ITR. Specifically, we first take the structure embeddings of attribute-missing samples

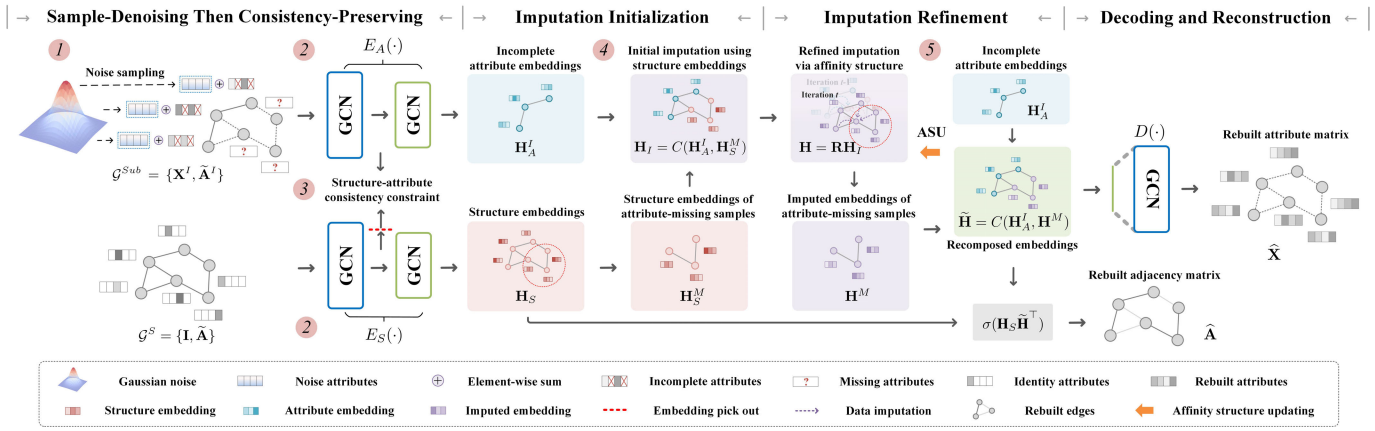


Fig. 2. Architecture of the RITR framework. To impute the incomplete values, we first initialize the original incomplete attributes as Gaussian noise for denoising learning (i.e., step 1), and then introduce a structure-attribute consistency constraint to refine the incomplete values by approximating a structure-attribute correlation matrix to a high-order structure matrix (i.e., step 3). To impute the missing values, we first adopt the structure embeddings of the attribute-missing samples as the embedding initialization (i.e., step 4), and then adaptively refine these initial values by aggregating the reliable and informative information of the attribute-incomplete samples according to the dynamic affinity structure (i.e., step 5).

as initial imputed variables, and then refine them with an adaptively updated affinity structure for embedding refinement. The above operations correspond to steps 4 and 5 in Fig. 2.

This work is a substantially extended version of our original conference paper [15]. Compared with its previous version, it has the following significant improvements.

1) *Novel Research Problem:* To the best of our knowledge, hybrid-absent graph machine learning is a rarely explored research field, yet it is a real-world demand from various applications. Accordingly, we develop a novel graph machine learning framework called RITR, which is the first incomplete and missing graph imputation network to solve the corresponding learning problem.
2) *Newly Proposed Mechanism:* To complete attribute-incomplete samples, we propose a new feature completion mechanism termed STC by following the initializing-then-refining imputation criterion. This operation encourages the model to generate more discriminative features for attribute-incomplete samples, which serves the subsequent attribute-missing imputation task better.
3) *More Experimental Results and Analyses:* Besides more discussions and extensions, comprehensive experiments on six benchmark datasets have been conducted to verify the effectiveness and superiority of our proposed methods in different absent graph situations.

The remainder of this article is organized as follows. Section II reviews related work in terms of unsupervised graph machine learning and graph machine learning on absent graphs. Section III presents the notations, definitions, model design, training objectives, and complexity analysis. Section IV presents the experimental results with the corresponding discussions. Section V draws a final conclusion.

II. Related Work

A. Unsupervised Graph Machine Learning

Early solutions to unsupervised graph machine learning mainly focus on random-walk-based methods [16], [17], which first generate the random walk sequences over the network structure properties and then use a Skip-Gram model to learn graph representations. However, these methods heavily rely on structure information and overlook other available properties (e.g., attribute information) in the graph. More recently, since the powerful neighborhood aggregation capacity of GNNs, many efforts have been made to design GNN-based methods [18], [19], [20]. As one of the most representatives, generative or predictive learning-oriented methods strive to harness the wealth of information embedded in data using well-established techniques, such as autoencoder learning [21], [22], [23], [24], [25] and adversarial learning [26], [27]. Another line pays attention to graph contrastive learning, which aims to maximize the agreement of two jointly sampled positive pairs [28], [29], [30], [31], [32], [33]. One underlying assumption commonly adopted by these methods is that the attributes of all the nodes are complete. While in real-world scenarios, they may suffer from significant performance degradation when encountering absent graphs.

B. Graph Machine Learning on Absent Graphs

According to the type of absent graphs, the existing absent graph machine learning methods could be roughly grouped into the following three categories.

1) Attribute-Incomplete Graph Machine Learning: In the attribute-incomplete circumstance, some methods propose to leverage data imputation-oriented techniques to restore the incomplete information, such as matrix completion [8], GAN [34], and GMM [35]. For instance, NMTR [36] and GRAPE [37], two typical matrix completion methods, first take the user–item rating matrix, users (or items), and the observed ratings as a bipartite graph, sample attributes, and connected relationships, respectively. These methods then use a GNN to predict the probabilities, which are considered as imputed values, for the missing connected relationships. Similarly, given a created bipartite graph, GRAPE [37] first converts a data imputation process into a linkage prediction learning task, and then uses a GNN to solve it. Recent efforts such as NMTR [36] and IGMC [9] follow the same paradigm

as previous matrix completion methods to conduct the data imputation and sample embedding in a transductive or inductive learning manner. In addition, GINN [10] first initializes the incomplete values by a binary mask matrix before network training, and then learns a GNN with an adversarial learning mechanism to complete the absent information. GCNMF [11] uses a GMM to estimate the incomplete features based on the available information. Meanwhile, it jointly optimizes the GMM and GNN in a united framework. More recently, T2-GNN [38] is a general teacher–student graph learning framework to restore both incomplete node features and graph structure through self-distillation.

2) Attribute-Missing Graph Machine Learning: Compared with the attribute-incomplete circumstance, handling the graph data with a majority of samples having no attributes poses more challenges in learning high-quality representations. This topic has attracted great attention from graph machine learning researchers recently. For example, HGNN-AC [39] first uses heterogeneous information networks (HINs) to learn representations and subsequently uses the topological relationship between nodes as guidance to implement feature completion for attribute-missing samples via an attention mechanism. HGCA [40] uses the contrastive learning technique to unify the processes of feature completion and representation learning, and thereafter, conduct a fine-grained attribute completion by extracting the semantic relationships among different types of samples. Besides the attribute-missing heterogeneous graph machine learning, an advanced method called SAT [14] makes the first attempt to solve the attribute-missing learning problem over the homogeneous graphs. By unifying the data imputation and network learning processes into a single optimization procedure, SAT learns two-source information embedding matrices in a decoupled manner and then aligns them with a noise matrix sampled from a normal distribution for data completion. Another recent work, ITR [15] introduces an initializing-then-refining mechanism, encouraging the network to fully use the trustworthy visible information to adaptively conduct the sample embedding for missing attribute imputation. FP [41] diffuses the features from observed nodes to neighbors whose features are incomplete based on the heat diffusion function. PSGNN [42] studies the usage of artificial positional node features and structural node features to help GNNs learn useful information from nonattributed graphs. More recently, SVGA [43] and Amer [44] develop an autoencoder-style framework to estimate missing node features via structured variational inference and adversarial learning techniques, respectively.

3) Hybrid-Absent Graph Machine Learning: As aforementioned, attribute-incomplete and attribute-missing graph machine learning problems have been intensively studied in recent years. Despite their significant progress, in nature, these methods are not capable of effectively handling hybrid-absent graphs. In this circumstance, especially for the unsupervised scenario, the performance of the existing attribute-incomplete and attribute-missing methods could drop drastically since they suffer from at least one of the following limitations: 1) heavily relying on annotated graph data; 2) lacking a specialized feature completion mechanism for handling attribute-missing or attribute-incomplete samples; 3) disconnecting the processes of data imputation and network optimization; 4) isolating the learning processes of structure and attribute embeddings; and 5) imposing too strict a distribution assumption on the latent variables. Although our recently proposed ITR could address most of the above problems and exhibits powerful learning capacity in the attribute-missing situation, it is still a great challenge to recover incomplete and missing values with limited available information simultaneously. To achieve this goal, we study a new important research problem termed hybrid-absent graph machine learning and design an ITR variant called RITR. It first leverages the intimate structure-attribute relationship to guide the imputation of incomplete attributes and then uses the most trustworthy visible information to implement the missing attribute completion. To the best of our knowledge, none of the above literature considers the hybrid-absent graph machine learning problem, and RITR is the first work dedicated to this field.

III. Method

A. Notations and Definitions

We denote $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ as a given undirected graph that contains N samples with C categories, where $\mathcal{V}$ and $\mathcal{E}$ indicate the node set and edge set, respectively. Generally, the topology of a graph $\mathcal{G}$ can be characterized by its adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ and the content of graph $\mathcal{G}$ can be represented by its attribute matrix $\mathbf{X} \in \mathbb{R}^{N\times D}$, where D refers to the sample dimension. The main notations and their explanations are summarized in Nomenclature.

Definition 1 (Hybrid-Absent Graph): We denote a hybrid-absent graph $\widetilde{\mathcal{G}} = \{\mathcal{V}^I, \mathcal{V}^M, \mathcal{E}\}$, where partial attributes of some samples are unavailable (i.e., the attribute-incomplete sample set $\mathcal{V}^I$) and all the attributes of other samples are entirely missing (i.e., the attribute-missing sample set $\mathcal{V}^M$). $N^I = |\mathcal{V}^I|$ and $N^I = |\mathcal{V}^M|$ refer to the number of attribute-incomplete samples and attribute-missing samples, respectively. Accordingly, $\mathcal{V} = \mathcal{V}^I \cup \mathcal{V}^M$, $\mathcal{V}^I \cap \mathcal{V}^M = \varnothing$, and $N = N^I + N^M$. Note that the structure information (i.e., $\mathcal{E}$) of $\widetilde{\mathcal{G}}$ is complete.

Definition 2 (Learning Task): In this work, we mainly focus on addressing the hybrid-absent graph machine learning problem on graphs without label annotation. Our autoencoder-style framework works for learning two graph encoding functions $E_A(\cdot)$ and $E_s(\cdot)$ to impute invisible latent variables. Then, a graph decoding function $D(\cdot)$ will recover the attribute-missing and attribute-incomplete samples based on the imputed hidden features. The recovered attributes can be saved and used for profiling and node classification tasks.

B. Overview

In the hybrid-absent graph scenarios, a good graph imputation model should fully leverage the available visible information to perform data completion. To achieve this goal, we design an STC mechanism and an ITR mechanism, which are intended to solve the attribute-incomplete and attribute-missing learning problems, respectively. As illustrated in Fig. 2, in the STC mechanism, we first generate a corrupted subgraph by randomly adding Gaussian noise

information to incomplete attributes as initial values. Then, the corrupted subgraph and the structural graph are transferred into two graph encoders to learn low-dimensional representations. In the information extraction phase, a structure-attribute consistency constraint allows the intermediate structure-attribute representations of attribute-incomplete samples to negotiate with each other to refine incomplete attributes. After that, the ITR mechanism uses the structure embeddings of the attribute-missing samples as the embedding initialization, and then adaptively refines these initial values by aggregating the reliable information of attribute-incomplete samples according to an affinity structure. By combining these two mechanisms, we expect that the proposed RITR can effectively improve the quality of data completion for hybrid-absent graphs. Finally, RITR conducts the graph reconstruction based on the graph embedding by jointly minimizing three objectives.

C. Sample-Denoising Then Consistency-Preserving

The critical technique extension of RITR against its conference version is the STC mechanism. This mechanism focuses on ensuring high-quality representations for nodes with incomplete attributes, which in turn offers a feasible initialization to those nodes with missing attributes in the subsequent steps.

Since the network could not be optimized over unknown values, we use a sample-denoising learning approach to ease the network training and facilitate the robustness of the learned attribute-incomplete features. Specifically, we first randomly generate a Gaussian noise matrix $\mathbf{N} \in \mathbb{R}^{N^I \times D}$ with iterations, and then assign it to the incomplete attribute matrix $\mathbf{X}^I \in \mathbb{R}^{N^I \times D}$ as initial values. The resultant matrix is denoted as a corrupted incomplete attribute matrix $\widetilde{\mathbf{X}}^I \in \mathbb{R}^{N^I \times D}$. Although the sample-denoising scheme has been previously explored and proved to be powerful [45], directly applying it to the restoration of incomplete attributes is less comprehensive since partial attributes are invisible. Inspired by the principle that attribute and structure information possess consistent and complementary properties of a typical graph [46], we leverage the intimate relationship between structure-attribute embeddings to enhance the initial imputation quality of incomplete attributes. To be specific, we first use two graph convolution-network-based encoders denoted as $E_A(\cdot)$ and $E_S(\cdot)$ to extract the latent features of attribute-incomplete samples and graph structure, respectively. Formally, $E_A(\cdot)$ accepts a subgraph $\mathcal{G}^{\text{sub}}$ with $\widetilde{\mathbf{X}}^I$ and its normalized adjacency matrix $\widetilde{\mathbf{A}}^I \in \mathbb{R}^{N^I \times N^I}$ as input and the lth representations of attribute-incomplete samples can be formulated as below

$$\mathbf{H}_A^{I(l)} = \sigma\left(\widetilde{\mathbf{A}}^I \mathbf{H}_A^{I(l-1)} \Theta^{(l)}\right) \tag{1}$$

where $\Theta^{(l)}$ is the parameter matrix of $E_A(\cdot)$ in the lth layer, and $\sigma(\cdot)$ indicates a nonlinear activation function. Similarly, $E_S(\cdot)$ receives a structure graph $\mathcal{G}^S$ with an identity matrix $\mathbf{I} \in \mathbb{R}^{N \times N}$ and a normalized adjacency matrix $\widetilde{\mathbf{A}} \in \mathbb{R}^{N \times N}$, and the lth representations of the graph structure can be obtained by calculating

$$\mathbf{H}_S^{(l)} = \sigma\left(\widetilde{\mathbf{A}} \mathbf{H}_S^{(l-1)} \Psi^{(l)}\right) \tag{2}$$

where $\Psi^{(l)}$ is the parameter matrix of $E_S(\cdot)$ in the lth layer. After that, we encourage each attribute-incomplete sample to be close to its counterpart and o-order neighbors across structure-attribute modalities, which can be formulated as

$$\mathcal{L}_C = \underbrace{\frac{1}{N^I}\sum_{i}^{N^I}(\mathbf{C}_{ii}-1)^2}_{\text{Self-loop consistency}} + \underbrace{\frac{1}{N^I(N^I-1)}\sum_{i}^{N^I}\sum_{j\neq i}^{N^I}\left(\mathbf{C}_{ij}-\widetilde{\mathbf{A}}_{ij}^{I(o)}\right)^2}_{\text{High-order structural consistency}} \tag{3}$$

$$\mathbf{C} = \frac{\mathbf{H}_A^{I(1)} p\left(\mathbf{H}_S^{(1)}\right)^\top}{\left\|\mathbf{H}_A^{I(1)}\right\| \left\|p\left(\mathbf{H}_S^{(1)}\right)\right\|} \tag{4}$$

where $\mathbf{C} \in \mathbb{R}^{N^I \times N^I}$ and $\widetilde{\mathbf{A}}^{I(o)} \in \mathbb{R}^{N^I \times N^I}$ denote a structure-attribute correlation matrix and an o-order normalized adjacency matrix, respectively. In addition, $\mathbf{H}_A^{I(1)}$ and $\mathbf{H}_S^{(1)}$ indicate the embeddings of attribute-incomplete samples and graph structure in the first layer, respectively. $p(\cdot)$ is an embedding pick-out function [14].

As seen in (3), the first term aims to ensure that the diagonal elements of the structure-attribute correlation matrix are close to a specific value, thereby promoting consistency in the structure-attribute embeddings of each attribute-incomplete sample. Moreover, the second term encourages each sample to be closer to its o-order neighbors than nonneighbors across both structure and attribute modalities. This objective aims to leverage diverse complete structure properties to facilitate the feature completion of attribute-incomplete samples. By doing this, both sample denoising and structure-attribute consistency constraint are seamlessly integrated to make the learned representations of attribute-incomplete samples robust and invariant to data perturbations (e.g., the noise and incompleteness of the graph). The overall pipeline of training the proposed STC is summarized in Algorithm 1.

Algorithm 1 Learning Procedure of the STC Mechanism

Require: Subgraph $\mathcal{G}^{\text{Sub}} = \{\mathbf{X}^I, \widetilde{\mathbf{A}}^I\}$; structure graph $\mathcal{G}^S = \{\mathbf{I}, \widetilde{\mathbf{A}}\}$; o-order normalized adjacency matrix of subgraph $\widetilde{\mathbf{A}}^{I(o)}$; maximum iterations T; hyperparameter β.
Ensure: Attribute embeddings of incomplete samples $\mathbf{H}_A^I$ and structure embeddings $\mathbf{H}_S$.
1: **for** $t = 1$ to T **do**
2: Generate $\mathbf{N}$ and assign it to $\mathbf{X}^I$;
3: Use $E_A(\cdot)$ to learn $\mathbf{H}_A^I$ by (1);
4: Use $E_S(\cdot)$ to learn $\mathbf{H}_S$ by (2);
5: Calculate $\mathbf{C}$ by (4)
6: Optimize the model by minimizing (3);
7: **end for**
8: **return** $\mathbf{H}_A^I$ and $\mathbf{H}_S$

D. Initializing Then Refining

The ITR mechanism is designed to fully use the trustworthy visible structure and attribute information produced by the STC mechanism for initializing and refining the missing values. It consists of the following two steps.

1) Imputation Initialization: The widely adopted measure for missing data initialization are the traditional imputation techniques, such as zero value filling and mean value filling. Nevertheless, in the attribute-missing circumstance, these filling methods could incorporate amounts of irrelevant noise

TABLE I
PERFORMANCE COMPARISON. BOTH ATTRIBUTE-INCOMPLETE AND ATTRIBUTE-MISSING RATIOS ARE SET TO 60%. OURS-Z, OURS-S, AND OURS-S-A ARE METHODS WHERE WE IMPUTE THE LATENT VARIABLES OF ATTRIBUTE-MISSING SAMPLES WITH ZERO VALUES, THE STRUCTURE EMBEDDINGS MERELY, AND THE STRUCTURE-ATTRIBUTE EMBEDDINGS, RESPECTIVELY. ↑ DENOTES THE PERFORMANCE IMPROVEMENT OF OURS-S-A AGAINST OURS-S

Dataset	Method	Recall@50	NDCG@50
Cora	Ours-Z	27.85	25.68
	Ours-S	28.59	26.45
	Ours-S-A	**32.43 (3.84 ↑)**	**30.88 (4.43 ↑)**
Citeseer	Ours-Z	18.56	18.56
	Ours-S	19.79	19.38
	Ours-S-A	**22.14 (2.35 ↑)**	**23.67 (4.29 ↑)**

that will diffuse through the network, causing semantically biased representations. To alleviate this issue, it is intuitive to leverage the structure embeddings as the embedding initialization for latent variables of attribute-missing samples. The reason for that is twofold. First, the attribute embedding and the structure embedding describe different aspects of a node, providing consistent and complementary information in these two modalities [46]. Second, this initialization approach is reliable since the structure information of the original graph is complete.

To this end, we first pick out the structure embeddings of attribute-missing samples $\mathbf{H}_S^M \in \mathbb{R}^{N^M \times d}$ from $\mathbf{H}_S$, and then use a Concat function $C(\cdot)$ to integrate $\mathbf{H}_S^M$ with $\mathbf{H}_A^I$, where d refers to the embedding dimension. It is worth noting that the information concatenation to construct $\mathbf{H}_I \in \mathbb{R}^{N \times d}$ is not the classic channelwise or rowwise concatenation. In this operation, the latent variables of attribute-incomplete samples are filled with $\mathbf{H}_A^I$ and the latent variables of attribute-missing samples are filled with $\mathbf{H}_S^M$

$$\mathbf{H}_I = C\left(\mathbf{H}_A^I, \mathbf{H}_S^M\right) \tag{5}$$

where $\mathbf{H}_I$ indicates the initially imputed embeddings. In our concatenation settings, the location of each sample remains unchanged within the original graph.

2) Imputation Refinement: As known, the trustworthiness degrees of attribute and structure information exhibit differences to some extent. Making full use of the trustworthy visible structure and attribute information to complete the missing values could boost the imputation quality of attribute-missing samples. To illustrate whether our argument holds, we make a comparison among three methods and discuss the performance on Cora and Citeseer. Here, we set both the attribute-incomplete and attribute-missing ratios as 60%. As seen in Table I, we can observe that: 1) Ours-S performs better than Ours-Z, verifying that the structure embeddings $\mathbf{H}_S^M$ can provide an effective embedding initialization and 2) the trustworthy attributes can provide more discriminative information to assist in the refinement of initial imputation. In our design, we leverage available attribute properties $\mathbf{H}_A^I$ to refine the initially imputed variables $\mathbf{H}_S^M$ via a dynamic affinity structure $\mathbf{R} \in \mathbb{R}^{N \times N}$, which can be written as a graph-convolution-like formulation

$$\mathbf{H} = \mathbf{R}\mathbf{H}_I \tag{6}$$

where $\mathbf{H} \in \mathbb{R}^{N \times d}$ indicates the imputed embeddings, and we initialize the affinity structure $\mathbf{R}$ as $\widetilde{\mathbf{A}}$.

According to (6), the attribute-missing imputation could be refined from the following two aspects. On one hand, it is obvious that the noise information in $\mathbf{H}_S^M$ can be transferred into the well-learned attribute embeddings of attribute-incomplete samples. This would undermine the representation quality and the reconstruction accuracy of available information, which in turn negatively affects the subsequent data imputation tasks and even distorts the original graph. To tackle this problem, we implement an information recomposing (IR) scheme to decrease the adverse effect of noise information passing from the embeddings of attribute-missing samples. First, we pick out the latent variables of attribute-missing samples $\mathbf{H}^M \in \mathbb{R}^{N^M \times d}$ from $\mathbf{H}$, and next recompose them with $\mathbf{H}_A^I$ using a Concat function $C(\cdot)$

$$\widetilde{\mathbf{H}} = C\left(\mathbf{H}_A^I, \mathbf{H}^M\right) \tag{7}$$

where $\widetilde{\mathbf{H}} \in \mathbb{R}^{N \times d}$ indicates the sample-recomposed embeddings. The IR scheme replaces the adjusted embeddings of attribute-incomplete samples with more reliable $\mathbf{H}_A^I$. Meanwhile, as illustrated in Fig. 2, we fix the embeddings of attribute-incomplete samples as $\mathbf{H}_A^I$ in the final step. This provides the most trustworthy information on attribute-incomplete samples for subsequent missing attribute imputation.

On the other hand, we argue that the initial affinity matrix $\mathbf{R}$ (i.e., $\widetilde{\mathbf{A}}$) is not the ground truth. The limitations within this matrix are twofold.

1) *Noisy Connections:* Besides inner connections within clusters, inappropriate connections could exist between clusters in the matrix.
2) *Missing Connections:* In $\widetilde{\mathbf{A}}$, only the first-order connections are preserved, and the high-order relevant connections could be missing.

Both would cause inaccurate imputation and reconstruction of missing attributes. To overcome these issues, we seek to refine $\mathbf{R}$ by emphasizing the dependable connections while weakening the unreliable ones. To this end, we propose an affinity structure updating (ASU) scheme to optimize $\mathbf{R}$ with iterations. Specifically, we first calculate a normalized self-correlated matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ according to $\widetilde{\mathbf{H}}$ as below

$$\mathbf{S}_{jk} = \mathcal{N}\left(\frac{\widetilde{\mathbf{h}}_j \widetilde{\mathbf{h}}_k^\top}{\|\widetilde{\mathbf{h}}_j\| \|\widetilde{\mathbf{h}}_k\|}\right) \quad \forall\ j, k \in [1, N] \tag{8}$$

where $\mathcal{N}(\cdot)$ indicates a structural normalization function. $\widetilde{\mathbf{h}}_j$ and $\widetilde{\mathbf{h}}_k$ indicate the embeddings of node $\mathbf{v}_j$ and node $\mathbf{v}_k$, respectively. Then we optimize the affinity structure $\mathbf{R}$ every t iterations via (9) and leverage it as guidance for the subsequent missing attribute imputation

$$\mathbf{R} = \gamma \widetilde{\mathbf{A}} + (1 - \gamma)\mathbf{S} \tag{9}$$

where γ is a balanced hyperparameter and is initialized as 0.5. With the ASU scheme, the network is enabled to construct the embeddings of attribute-missing samples with not only the first-order but also the trustworthy high-order connections within the graph structure. As the embeddings of attribute-incomplete samples become more reliable and the embeddings of attribute-missing samples become more

TABLE II
COMPARISON OF THE COMPLEXITY ANALYSIS AMONG SAT [14], ITR [15], AND THE PROPOSED RITR

Model	Time complexity (model and loss function computation)
SAT	$\mathcal{O}(Nd(D_A+D_S+dL)+\lvert\mathcal{E}\rvert dL+N^2+nD_A)\approx\mathcal{O}(N^2)$
ITR	$\mathcal{O}(Nd(D_A+D_S+dL)+\lvert\mathcal{E}\rvert(dL+D_A)+N^2+nD_A)\approx\mathcal{O}(N^2)$
RITR	$\mathcal{O}(Nd(D_A+D_S+dL)+\lvert\mathcal{E}\rvert(dL+D_A)+N^2+n(n+D_A))\approx\mathcal{O}(N^2)$

Algorithm 2 Learning Procedure of the ITR Mechanism

Require: Attribute embeddings of incomplete samples $\mathbf{H}_A^I$; structure embeddings $\mathbf{H}_S$; maximum iterations T; update interval U; elementwise indicator matrix $\mathbf{M}$; hyperparameters γ and α.
Ensure: Rebuilt attribute matrix $\widehat{\mathbf{X}}$ and adjacency matrix $\widehat{\mathbf{A}}$.
1: **for** $t=1$ to T **do**
2: Obtain $\mathbf{H}_S^M$ from $\mathbf{H}_S$ in accordance with indices;
3: Combine $\mathbf{H}_A^I$ with $\mathbf{H}_S^M$ for initial imputation by (5);
4: Conduct the imputation refinement using $\mathbf{R}$ by (6);
5: Recompose the latent variables to obtain $\widetilde{\mathbf{H}}$ by (7);
6: Calculate $\mathbf{S}$ over the resultant $\widetilde{\mathbf{H}}$ by (8);
7: **if** $t\%U==0$ **then**
8: Update $\mathbf{R}$ to refine $\mathbf{H}$ by (9) and (6);
9: **end if**
10: Use $D(\cdot)$ to decode $\widetilde{\mathbf{H}}$ and output $\widehat{\mathbf{X}}$ by (10);
11: Use a simple inner decoder to rebuild $\widehat{\mathbf{A}}$;
12: Optimize the model by minimizing (11) and (12);
13: **end for**
14: **return** $\widehat{\mathbf{X}}$ and $\widehat{\mathbf{A}}$

informative, the learned representations can exhibit enhanced discriminative and robust characteristics. The overall training pipeline for the proposed ITR is summarized in Algorithm 2.

E. Training Objectives and Complexity Analysis

1) Training Objectives: After obtaining $\widetilde{\mathbf{H}}$, we feed it with $\widetilde{\mathbf{A}}$ into a graph decoder $D(\cdot)$ to rebuilt the attributes of attribute-incomplete and attribute-missing samples

$$\widetilde{\mathbf{H}}^{(l)}=\sigma\left(\widetilde{\mathbf{A}}\widetilde{\mathbf{H}}^{(l-1)}\Phi^{(l)}\right) \tag{10}$$

where $\Phi^{(l)}$ indicates the parameter matrix of $D(\cdot)$ in the lth layer. $\widetilde{\mathbf{H}}^{(0)}$ and $\widetilde{\mathbf{H}}^{(2)}$ denote the sample-recomposed embeddings $\widetilde{\mathbf{H}}$ and the rebuilt attribute matrix $\widehat{\mathbf{X}}\in\mathbb{R}^{N\times D}$, respectively. The joint loss function of RITR includes three parts, which can be written as

$$\mathcal{L}_A=\frac{1}{2N^I}\left\|\mathbf{M}\odot(\mathbf{X}^I-\widehat{\mathbf{X}}^I)\right\|_F^2 \tag{11}$$

$$\mathcal{L}_S=\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\mathrm{BCE}\left(\widetilde{\mathbf{A}}_{ij}\widehat{\mathbf{A}}_{ij}\right) \tag{12}$$

$$\mathcal{L}=\alpha\mathcal{L}_A+\mathcal{L}_S+\beta\mathcal{L}_C. \tag{13}$$

In (11), $\mathcal{L}_A$ refers to the mean square error (MSE) of attribute-incomplete samples between $\mathbf{X}$ and $\widehat{\mathbf{X}}$. $\mathbf{M}\in\mathbb{R}^{N^I\times D}$ is an elementwise indicator matrix where $\mathbf{M}_{ij}=1$ if $\mathbf{X}_{ij}^I$ is a real value, otherwise $\mathbf{X}_{ij}^I$ is a null value (i.e., an incomplete attribute). In (12), $\mathcal{L}_S$ refers to the binary cross-entropy (BCE) between $\widetilde{\mathbf{A}}$ and the rebuilt adjacency matrix $\widehat{\mathbf{A}}\in\mathbb{R}^{N\times N}$, where $\widehat{\mathbf{A}}=\sigma(\mathbf{H}_S\widetilde{\mathbf{H}}^\top)$, and $\sigma(\cdot)$ is a Sigmoid activation function. α and β are two balanced hyperparameters. The applied optimization objectives are similar to the existing attribute-missing graph machine learning methods [14], [15]. However, the major differences between current methods and RITR could be summarized in the following three parts: 1) more naturally handling hybrid-absent graphs in an unsupervised circumstance; 2) more comprehensive that seamlessly unifies the representation learning and data imputation processes of attribute-incomplete and attribute-missing samples into a common optimization framework; and 3) more discriminative that enables the structure-attribute information to sufficiently negotiate with each other for feature completion by performing STC and ITR mechanisms.

2) Complexity Analysis: The time complexity of the proposed RITR could be discussed from the following two aspects: the graph autoencoder framework and the loss function computation. For two GCN-based graph encoders, the complexities of $E_A(\cdot)$ and $E_S(\cdot)$ are $\mathcal{O}(Nd^2(L-1)+NdD_A+\lvert\mathcal{E}\rvert dL)$ and $\mathcal{O}(Nd^2(L-1)+NdD_S+\lvert\mathcal{E}\rvert dL)$, where N, L, and $\lvert\mathcal{E}\rvert$ are the number of nodes, encoder layers, and edges, respectively. D_A, D_S, and d are the dimensions of raw attribute features, raw structure features, and latent features, respectively. For the graph decoder, the complexity of $D(\cdot)$ is $\mathcal{O}(Nd^2(L-1)+NdD_A+\lvert\mathcal{E}\rvert d(L-1)+\lvert\mathcal{E}\rvert D_A)$. For the loss function computation, we follow SAT [14] and use the MSE and BCE loss functions to reconstruct node attributes and graph structure, respectively. The time complexities of $\mathcal{L}_A$ and $\mathcal{L}_S$ are $\mathcal{O}(nD_A)$ and $\mathcal{O}(N^2)$, respectively. The time complexity of the structure-attribute consistency loss function $\mathcal{L}_C$ is $\mathcal{O}(n^2)$, where $n=N-Nr$, in which r is the ratio of attribute-missing samples. Considering the attribute-missing problem is universal in real-world scenarios (i.e., r can be assigned a large value), and n^2 can be relatively small, thus the computation overhead in this context is deemed acceptable. The overall time complexity of RITR for each training iteration is $\mathcal{O}(Nd(D_A+D_S+dL)+\lvert\mathcal{E}\rvert(dL+D_A)+N^2+n(n+D_A))\approx\mathcal{O}(N^2)$. To ensure a fair comparison, we perform a complexity comparison among three attribute-missing graph machine learning methods. As shown in Table II, we observe that RITR does not introduce any additional computational complexity compared with its competitors.

IV. EXPERIMENTS

A. Experimental Setup

1) Benchmark Datasets: We implement experiments to evaluate two proposed methods (i.e., ITR and RITR), on six benchmark datasets, including Cora, Citeseer, Amazon Computer (Amac), Amazon Photo (Amap), Ogbn-arxiv (Ogbn-A), and Ogbn-products (Ogbn-Ps for abbreviation). We summarize the detailed dataset information in Table III.

1) Cora, Citeseer, and Ogbn-A are three popular citation network datasets. Specially, nodes mean scientific publications, and edges mean citation relationships.

TABLE III
SUMMARY OF DATASETS

Dataset	Nodes	Edges	Dimension	Classes
Cora	2,708	5,278	1,433	7
Citeseer	3,327	4,228	3,703	6
Amac	13,752	245,861	767	10
Amap	7,650	119,081	745	8
Ogbn-arxiv	169,343	1,166,243	128	40
Ogbn-products	2,449,029	61,859,140	100	47

Each node has a predefined feature with the corresponding dimensions.

2) Amap, Amac, and Ogbn-Ps are segments of the Amazon co-purchase network, where nodes represent goods, edges indicate that two goods are frequently bought together, node features are bag-of-words encoded product reviews, and class labels are given by the product category.

2) Implementation Procedures: Both ITR and RITR are implemented with the PyTorch platform. We evaluate the effectiveness of two proposed methods through a two-step learning procedure. First, we train an unsupervised framework to learn representations and complete absent information for at least 600 iterations. Following SAT [14], we regard the profiling learning as a pretext task and adopt Recall@K and NDCG@K as metrics to evaluate the quality of rebuilt attributes. To alleviate the overfitting problem, we perform an early stop strategy when the loss value reaches a plateau. Second, for the node classification task, we feed the rebuilt attribute matrix into a graph classifier, optimize it with fivefold validation ten times, and report the average accuracy (ACC) performance.

3) Training Settings: In the attribute-missing case, we record the performance of all the methods directly according to this article of SAT [14] except for GINN [10], GCNMF [11], and SVGA [43]. In the hybrid-absent case, we run the released source code of all the compared methods by following the settings of the corresponding literature and report their results. For our proposed ITR and RITR, we strictly follow the criterion of data splits as was done in SAT, including the split ratio of attribute-complete/missing samples and the split ratio of train/test sets. Specifically, the following conditions hold.

1) In the profiling task, we randomly sample 40% nodes with complete attributes as the training set and manually mask all the attributes of the rest of 10% and 50% nodes (i.e., attribute-missing samples) as the validation set and the test set, respectively. Besides, when attribute-incomplete and attribute-missing samples exist simultaneously within a graph, we randomly mask 60% attributes of each attribute-complete sample (i.e., the training set) before network learning. We use a four-layer graph autoencoder framework and update its parameters with the Adam optimization algorithm. During the training phase, we transfer all the samples into ITR and RITR to complete absent attributes by merely reconstructing the available ones. After training, we rebuild the attribute matrix over the well-trained model via forwarding propagation.
2) In the node classification task, we randomly split the rebuilt attributes into 80% and 20% for training and testing, respectively. We train the classifier with fivefold validation for 1000 iterations and repeat the experiments ten times. According to the results of parameter sensitivity testing, we fix two balanced hyperparameters α and β to 10. Moreover, the learning rate, the latent dimension, the dropout rate, and the weight decay are set to $1e{-}3$, 64, 0.5, and $5e{-}4$, respectively. Note that we do not carefully tune these parameters for ease of training.

4) Compared Methods: We compare RITR with 15 existing baseline methods for feature estimation on both the attribute-missing and hybrid-absent graphs. Specifically, NeighAggre (NAS' 08) [47] is a classical profiling method. VAE (NeurIPS' 16) [48] is a well-known autoencoder method. GCN (ICLR' 17) [49], GraphSage (NeurIPS' 17) [50], and GAT (ICLR' 18) [51] are three typical GNNs. GraphRNA (KDD' 19) [52] and ARWMF (NeurIPS' 19) [53] are representatives of attributed random-walk-based methods. Hers (AAAI' 19) [54] is a cold-start recommendation method. DCLN (TNNLS' 23) [55] and HSHSM (TNNLS' 23) [56] are two advanced self-supervised graph representation learning methods. SAT (TPAMI' 22) [14] is the first attribute-missing graph imputation network. SVGA (KDD' 22) [43] and ITR (IJCAI' 22) [15] are two most advanced attribute-missing graph autoencoders. GINN (NN' 20) [10] and GCNMF [10] (FGCS' 21) [11] are two state-of-the-art attribute-incomplete graph machine learning methods.

B. Attribute-Missing Scenario

1) Performance Comparison: As shown in Table IV, we report the profiling performance of all the methods mentioned above. This table shows that ITR and RITR outperform all the compared baseline methods in terms of six metrics on four datasets. Specifically, the following conditions hold.

1) We first compare NeighAggre and VAE with our methods. Instead of merely exploiting the structure or attribute information for data imputation, our methods have two-source information sufficiently negotiate with each other, thus consistently exceeding NeighAggre and VAE by a large margin.
2) ITR and RITR show superior performance against GCN, GraphSage, and GAT, all of which have demonstrated strong representation learning capability in handling attribute-complete graphs. However, the results imply that these methods are not suitable to solve the attribute-missing problem.
3) For the two strongest attribute-missing graph machine learning methods (i.e., SVGA and SAT), RITR outperforms them by 1.81%/3.40%, 1.23%/4.75%, 0.79%/2.20%, and 0.97%/1.13% in terms of NDCG@50 metric on four datasets, respectively. This is because these baselines heavily rely on predefined assumptions that may not always hold in real-world graphs for data imputation, while RITR does not make any prior distribution assumption so that it can flexibly and effectively make full use of visible information for feature completion.

TABLE IV

PROFILING PERFORMANCE COMPARISON IN THE ATTRIBUTE-MISSING CIRCUMSTANCE. THE ATTRIBUTE-MISSING RATIO IS SET TO 60%. THE BOLDFACE AND UNDERLINE VALUES INDICATE THE BEST AND THE SUBOPTIMAL RESULTS (%), RESPECTIVELY

Dataset	Metric	NeighAggre	VAE	GCN	GraphSage	GAT	Hers	GraphRNA	ARWMF	SVGA	SAT	ITR	RITR
Cora	Recall@10	9.06	8.87	12.71	12.84	13.50	12.26	13.95	12.91	16.06	15.08	16.82	**17.32**
	Recall@20	14.13	12.28	17.72	17.84	18.12	17.23	20.43	18.13	23.17	21.82	23.69	**24.47**
	Recall@50	19.61	21.16	29.62	29.72	29.72	27.99	31.42	29.60	35.83	34.29	36.47	**36.81**
	NDCG@10	12.17	12.24	17.36	17.68	17.91	16.94	19.34	18.24	23.19	21.12	23.20	**24.20**
	NDCG@20	15.48	14.52	20.76	21.02	20.99	20.31	23.62	21.82	26.81	25.46	27.81	**28.97**
	NDCG@50	18.50	19.24	27.02	27.28	27.11	25.96	29.38	27.76	33.71	32.12	34.60	**35.52**
Citeseer	Recall@10	5.11	3.82	6.20	6.12	5.61	5.76	7.77	5.52	8.83	7.64	9.63	**10.16**
	Recall@20	9.08	6.68	10.97	10.97	10.12	10.25	12.72	10.15	14.55	12.80	15.48	**15.97**
	Recall@50	15.01	12.96	20.52	20.58	19.57	19.73	22.71	19.52	25.91	23.77	26.84	**26.90**
	NDCG@10	8.23	6.01	10.26	10.03	8.78	9.04	12.91	8.59	15.21	12.98	16.32	**17.16**
	NDCG@20	11.55	8.39	14.23	13.93	12.53	12.79	17.03	12.45	20.08	17.29	21.20	**22.03**
	NDCG@50	15.60	12.51	20.49	20.34	18.72	19.00	23.58	18.58	27.99	24.47	28.69	**29.22**
Amac	Recall@10	3.21	2.55	2.73	2.69	2.71	2.73	3.86	2.80	3.97	3.91	4.48	**4.59**
	Recall@20	5.93	5.02	5.33	5.28	5.30	5.25	6.90	5.44	7.22	7.03	7.81	**8.06**
	Recall@50	13.06	11.96	12.75	12.78	12.78	12.73	14.65	12.89	15.75	15.14	16.20	**16.48**
	NDCG@10	7.88	6.32	6.71	6.64	6.73	6.76	9.31	6.94	10.11	9.63	10.95	**11.14**
	NDCG@20	11.56	9.70	10.27	10.20	10.28	10.25	13.33	10.53	14.98	13.79	15.37	**15.69**
	NDCG@50	19.23	17.21	18.24	18.22	18.30	18.25	21.55	18.51	23.84	22.43	24.29	**24.63**
Amap	Recall@10	3.29	2.76	2.94	2.95	294	2.92	3.90	2.94	4.14	4.10	4.36	**4.49**
	Recall@20	6.16	5.38	5.73	5.62	5.73	5.74	7.03	5.68	7.51	7.43	7.82	**8.00**
	Recall@50	13.61	12.79	13.24	13.22	13.24	13.28	15.08	13.27	15.93	15.97	16.43	**16.61**
	NDCG@10	8.13	6.75	7.05	7.12	7.05	7.14	9.59	7.27	10.21	10.06	10.73	**10.94**
	NDCG@20	11.96	10.31	10.82	10.79	10.83	10.94	13.77	10.98	14.73	14.50	15.33	**15.56**
	NDCG@50	19.98	18.30	18.93	18.96	18.92	19.06	22.32	19.15	23.75	23.59	24.50	**24.72**

TABLE V

NODE CLASSIFICATION PERFORMANCE COMPARISON. BOTH ATTRIBUTE-INCOMPLETE AND ATTRIBUTE-MISSING RATIOS ARE SET TO 60%. THE BOLDFACE AND UNDERLINE VALUES INDICATE THE BEST AND THE SUBOPTIMAL RESULTS (%), RESPECTIVELY

Dataset	X (attribute-missing scenario)				X+A (attribute-missing scenario)				X (hybrid-absent scenario)				X+A (hybrid-absent scenario)			
	Cora	Citeseer	Amac	Amap	Cora	Citeseer	Amac	Amap	Cora	Citeseer	Amac	Amap	Cora	Citeseer	Amac	Amap
GCN	39.43	37.68	36.60	26.83	43.87	40.79	39.74	36.56	29.39	20.96	35.76	25.70	30.40	26.89	38.88	36.08
GAT	41.43	21.29	37.47	25.98	45.25	26.88	40.34	37.89	29.39	21.26	36.98	25.39	30.80	26.49	39.78	37.75
DCLN	78.01	64.32	82.74	88.87	83.55	66.61	86.77	90.55	74.13	59.44	80.08	85.50	81.12	63.77	82.27	86.73
HCHSM	79.12	65.18	83.40	89.05	83.28	67.22	85.13	90.60	74.94	54.48	80.22	86.17	81.56	64.02	84.07	86.99
GINN	-	-	-	-	67.58	55.32	81.27	87.77	-	-	-	-	38.16	23.66	25.45	37.25
GCNMF	-	-	-	-	70.30	63.40	76.43	87.79	-	-	-	-	57.19	50.90	35.40	82.04
SVGA	78.70	62.33	72.56	88.55	83.78	66.19	85.87	89.90	67.91	54.51	64.44	82.49	77.70	60.30	80.56	84.33
SAT	76.44	60.10	74.10	87.62	83.27	65.99	85.19	91.63	72.53	54.15	68.94	84.53	80.37	64.16	83.33	90.49
ITR	81.43	67.15	83.88	90.75	85.56	68.09	87.65	91.87	75.04	61.05	80.68	88.98	82.53	64.93	84.96	88.98
RITR	**81.64**	**67.47**	**85.28**	**91.27**	**85.81**	**69.01**	**88.49**	**92.24**	**78.11**	**63.75**	**82.92**	**90.49**	**84.29**	**66.00**	**86.19**	**91.75**

4) RITR achieves better performance than ITR on all the datasets. The superior results of RITR over the state-of-the-art method further verify the effectiveness of our improved framework for handling attribute-missing graphs.

Moreover, we report the node classification performance of ten methods in Table V. "X" or "X+A" indicates that the classifier receives the attribute matrix or attribute and adjacency matrices as input in the node classification task. Note that here we only take the attribute-missing scenario into consideration. From these results, we can see that the following conditions hold.

1) The classification results of GINN and GCNMF are not comparable to those of our two methods. ITR and RITR achieve at least 15.26%/15.51%, 4.69%/5.61%, 6.38%/7.22%, and 4.08%/4.45 accuracy increment. This indicates that these attribute-incomplete methods fall into inaccurate data imputation with extremely limited observations so that they cannot learn effective representations.
2) Taking the performance of "X" for instance, ITR and RITR gain 4.99%/5.20%, 7.05%/7.37%, 9.78%/11.18%, and 3.13%/3.65% performance enhancement over the state-of-the-art SAT method.

Similar observations can be obtained among SVGA, ITR, and RITR. These benefits can be attributed to the following merits: 1) different from SVGA and SAT, our proposed graph imputation networks avoid the reliance on any prior distribution assumption for missing attribute completion, so that they can facilitate the structure-attribute negotiation more flexibly and comprehensively and 2) the trustworthy visible attribute information and structure information can be used unitedly by ITR and RITR for data imputation instead of being treated separately. The above experimental results well demonstrate the superiority of ITR and RITR in the attribute-missing scenario.

2) Effect of Two Schemes in ITR Criterion: To verify the benefit of the initializing-then-refining imputation criterion, we conduct ablation studies on four datasets to compare ITR and two ITR variants, each of which has one of the critical components removed. From the results in Fig. 3, we observe that the accuracy of ITR on four datasets would degrade without one of the key components. Specifically, for the "X" task, ITR exceeds ITR w/o IR by 2.54%, 1.53%, 2.24%, and 1.64% accuracy increment, and ITR w/o ASU by 0.69%, 0.30%, 0.98%, and 0.55% accuracy increment on Cora, Citeseer, Amac, and Amap, respectively. We find that the IR scheme plays a more important role than the information

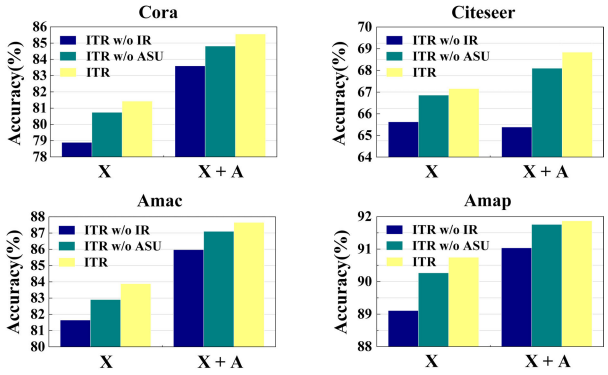


Fig. 3. Effect of the IR and ASU schemes for node classification. ITR w/o IR and ITR w/o ASU indicate the method with IR and ASU being masked, respectively.

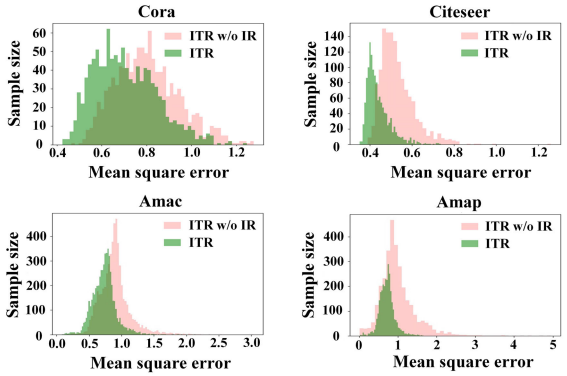


Fig. 4. MSE comparison of ITR and an ITR variant. ITR w/o IR denotes the ITR without IR scheme. The X- and Y-axis refer to the range of MSE and the sample size, respectively.

refining scheme. To illustrate this point visually, we present the MSE comparison of ITR and ITR w/o IR at the last training iteration. As seen in Fig. 4, the method with the IR scheme achieves a better convergence than ITR w/o IR. This indicates that our IR operation can effectively prohibit inaccurate information from being propagated, so the model can learn reliable representations for high-quality missing attribute restoration. All the above observations demonstrate the effectiveness of our proposed ITR imputation criterion, which can enable the structure-attribute information to sufficiently negotiate with each other for more accurate data imputation.

C. Hybrid-Absent Scenario

1) Performance Comparison: In this section, we further investigate the performance of our proposed methods and study a more challenging hybrid-absent problem, i.e., both attribute-missing and attribute-incomplete samples exist simultaneously within a graph. "X" or "X+A" indicates that the classifier receives the attribute matrix or attribute and adjacency matrices as input in the node classification task. To evaluate the quality of the rebuilt attribute matrix, we take eight methods (i.e., GCN, GAT, DCLN, HCHSM, GINN, GCNMF, SVGA, and SAT) as baselines and report the classification accuracy on four datasets. From these results in Table V, we can find that the following conditions hold.

1) In the "X" node classification task, although ITR outperforms all the baseline methods and achieves highly competitive results, it suffers from a significant average performance degradation of 4.37% in the hybrid-absent scenario compared with the attribute-missing scenario. This is because ITR conducts sample embedding over attribute-incomplete samples directly so that amounts of error information have diffused through the network. As a result, the resultant representations are inaccurate and can hardly provide the attribute-missing samples with discriminative enough information for feature completion.
2) Taking the "X" node classification task in the hybrid-absent scenario for example, RITR achieves the best performance against all the compared baselines. Specifically, RITR improves SVGA and SAT by 10.20%/5.58%, 9.24%/9.60%, 18.48%/13.98%, and 8.00%/5.96% accuracy increment on all the datasets. The observations of other cases are similar.

These results once again verify that when both the attribute-incomplete and attribute-missing samples exist simultaneously, the feature completion mechanisms of the existing baseline methods have an adverse effect on the quality of recovered attributes due to the propagation of incorrect information. In contrast, RITR effectively alleviates this adverse influence by introducing two personalized feature completion mechanisms based on the initializing-then-refining imputation criterion.

2) Effect of Each Mechanism of RITR: Here we conduct an ablation study to validate the effectiveness of our proposed attribute-incomplete and attribute-missing imputation mechanisms. Table VI reports the Recall and NDCG performance of three methods, including RITR w/o STC, RITR w/o ITR, and RITR. Specially, RITR w/o STC or RITR w/o ITR indicates that the method removes the STC mechanism or the ITR mechanism. Note that here we set both the attribute-missing and incomplete ratios as 60%. Table VI reports the results of RITR and its two variants, from which we can see that the following conditions hold.

1) RITR consistently improves RITR w/o STC on all the datasets. Taking the results on Cora for instance, RITR gains 0.73%, 1.02%, 1.52%, 1.06%, 1.22%, and 1.45% increment in terms of Recall and NDCG, demonstrating the effectiveness of leveraging the intimate structure-attribute relationship to guide the imputation of incomplete attributes. Similar observations can be concluded from the results on other datasets.
2) RITR significantly outperforms RITR w/o ITR and has performance enhancements of 4.73%, 3.97%, 2.80%, and 2.54% over it in terms of Recall@50 on four datasets, respectively. These results imply the importance of an effective imputation strategy in which we use the most trustworthy visible information to implement the missing attribute completion. In summary, this ablation study clearly validates that each mechanism can contribute to the overall performance of RITR.

3) Analysis of the AIR: To further investigate the superiority of RITR, it is necessary to show whether the proposed RITR can still achieve effective feature completion when less visible attribute information is available. To this end, we make a

TABLE VI

ABLATION STUDY ON THE EFFECTIVENESS OF TWO MECHANISMS OF RITR. RITR W/O STC AND RITR W/O ITR INDICATE THAT THE METHOD WITHOUT THE STC MECHANISM AND THE ITR MECHANISM, RESPECTIVELY. BOTH ATTRIBUTE-INCOMPLETE AND ATTRIBUTE-MISSING RATIOS ARE SET TO 60%. THE BOLDFACE AND UNDERLINE VALUES INDICATE THE BEST AND THE SUBOPTIMAL RESULTS (%), RESPECTIVELY

Dataset	Method	Recall@10	Recall@20	Recall@50	NDCG@10	NDCG@20	NDCG@50
Cora	RITR w/o STC	13.63	19.67	30.91	19.45	23.47	29.43
	RITR w/o ITR	12.03	16.52	27.70	16.79	19.84	25.66
	RITR	**14.36**	**20.69**	**32.43**	**20.51**	**24.69**	**30.88**
Citeseer	RITR w/o STC	7.08	11.93	21.61	12.12	16.37	22.80
	RITR w/o ITR	5.19	9.30	18.17	8.03	11.46	17.25
	RITR	**7.78**	**12.51**	**22.14**	**13.38**	**17.33**	**23.67**
Amac	RITR w/o STC	3.92	6.97	15.10	9.84	14.05	22.72
	RITR w/o ITR	2.83	5.49	12.92	7.01	10.63	18.60
	RITR	**4.26**	**7.47**	**15.72**	**10.44**	**14.73**	**23.50**
Amap	RITR w/o STC	3.96	7.16	15.58	10.04	14.35	23.33
	RITR w/o ITR	3.04	5.79	13.46	7.53	11.25	19.47
	RITR	**4.25**	**7.50**	**16.00**	**10.42**	**14.78**	**23.84**

TABLE VII

PERFORMANCE COMPARISON BETWEEN SAT AND OUR PROPOSED RITR. WE FIX THE ATTRIBUTE-MISSING RATIO AS 60% AND VARY THE AIR FROM 10% TO 70%. THE BOLDFACE VALUES INDICATE THE BEST RESULTS (%)

Dateset	Cora													
AIR	10%		20%		30%		40%		50%		60%		70%	
Method	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR
Recall@10	14.74	**17.12**	14.84	**16.59**	14.23	**16.08**	13.84	**15.86**	13.59	**14.95**	12.62	**14.36**	12.34	**13.16**
Recall@20	21.45	**23.72**	21.46	**23.36**	20.70	**23.02**	20.17	**22.29**	19.57	**21.40**	18.46	**20.69**	16.78	**18.72**
Recall@50	34.08	**36.32**	34.00	**35.76**	33.06	**35.19**	32.34	**34.38**	31.48	**33.38**	29.97	**32.43**	27.75	**30.21**
NDCG@10	20.70	**23.94**	20.73	**23.35**	20.12	**22.66**	19.50	**22.37**	19.12	**21.37**	18.01	**20.51**	16.83	**19.08**
NDCG@20	25.11	**28.40**	25.09	**27.86**	24.41	**27.26**	23.74	**26.67**	23.11	**25.69**	21.91	**24.69**	19.83	**22.80**
NDCG@50	31.80	**35.10**	31.78	**34.44**	30.90	**33.73**	30.15	**33.06**	29.40	**32.04**	27.94	**30.88**	25.62	**28.83**
Dateset	Citeseer													
AIR	10%		20%		30%		40%		50%		60%		70%	
Method	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR
Recall@10	7.27	**9.69**	7.23	**9.48**	7.24	**9.09**	6.95	**8.79**	6.61	**8.44**	5.84	**7.78**	5.68	**7.19**
Recall@20	12.33	**15.50**	11.95	**15.12**	11.94	**14.59**	11.52	**14.00**	11.03	**13.44**	9.95	**12.51**	9.97	**11.54**
Recall@50	22.90	**26.36**	22.28	**25.87**	22.55	**25.16**	21.75	**24.23**	20.88	**23.46**	19.20	**22.14**	18.99	**20.58**
NDCG@10	12.61	**16.50**	12.40	**16.16**	12.25	**15.59**	11.85	**15.11**	11.29	**14.47**	9.71	**13.38**	8.63	**12.51**
NDCG@20	16.83	**21.37**	16.34	**20.88**	16.17	**20.18**	15.65	**19.46**	14.98	**18.64**	13.13	**17.33**	12.21	**16.14**
NDCG@50	23.74	**28.50**	23.09	**27.94**	23.11	**27.13**	22.35	**26.18**	21.41	**25.21**	19.19	**23.67**	18.10	**22.06**
Dateset	Amac													
AIR	10%		20%		30%		40%		50%		60%		70%	
Method	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR
Recall@10	3.81	**4.50**	3.81	**4.47**	3.78	**4.44**	3.75	**4.39**	3.70	**4.36**	3.59	**4.26**	3.24	**4.14**
Recall@20	6.88	**7.88**	6.82	**7.83**	6.79	**7.76**	6.70	**7.69**	6.69	**7.60**	6.52	**7.47**	6.13	**7.39**
Recall@50	14.86	**16.24**	14.79	**16.21**	14.73	**16.10**	14.64	**16.02**	14.52	**15.91**	14.28	**15.72**	13.99	**15.83**
NDCG@10	9.38	**10.97**	9.37	**10.94**	9.32	**10.87**	9.26	**10.75**	9.09	**10.65**	8.86	**10.44**	8.75	**10.17**
NDCG@20	13.49	**15.45**	13.42	**15.39**	13.36	**15.29**	13.23	**15.14**	13.09	**14.98**	12.79	**14.73**	12.40	**14.53**
NDCG@50	22.00	**24.33**	21.92	**24.30**	21.84	**24.16**	21.70	**23.99**	21.47	**23.81**	21.11	**23.50**	20.91	**23.53**
Dateset	Amap													
AIR	10%		20%		30%		40%		50%		60%		70%	
Method	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR	SAT	RITR
Recall@10	3.95	**4.40**	3.90	**4.37**	3.92	**4.35**	3.90	**4.30**	3.86	**4.27**	3.82	**4.25**	3.53	**4.12**
Recall@20	7.16	**7.86**	7.13	**7.82**	7.13	**7.76**	7.05	**7.71**	6.97	**7.64**	6.93	**7.50**	6.43	**7.26**
Recall@50	15.49	**16.48**	15.42	**16.44**	15.44	**16.36**	15.34	**16.29**	15.16	**16.19**	15.20	**16.00**	14.14	**15.39**
NDCG@10	9.68	**10.80**	9.60	**10.75**	9.63	**10.67**	9.57	**10.57**	9.48	**10.48**	9.39	**10.42**	8.71	**10.14**
NDCG@20	13.98	**15.39**	13.92	**15.33**	13.93	**15.22**	13.81	**15.11**	13.65	**14.96**	13.57	**14.78**	12.62	**14.34**
NDCG@50	22.86	**24.55**	22.76	**24.50**	22.80	**24.37**	22.66	**24.25**	22.41	**24.09**	22.39	**23.84**	20.89	**23.00**

performance comparison between SAT and RITR by varying the attribute-incomplete ratio (AIR) from 10% to 70% and fixing the attribute-missing ratio as 60%. From the results in Table VII, several observations can be summarized as follows.

1) RITR consistently performs better than SAT in all the situations on four datasets. For example, RITR outperforms SAT by 2.42%, 2.25%, 1.85%, 1.84%, 1.83%, 1.94%, and 1.51% in terms of Recall@10 when the AIR varies from 10% to 70% on Citeseer. The observations of other metrics and datasets are similar. This is because the SAT method cannot implement an effective latent distribution matching between the embeddings of incomplete node attributes and graph structure. Naturally, the resultant misleading information poses a negative impact on data imputation and feature completion, resulting in suboptimal representations. RITR can effectively model hybrid-absent graphs and alleviate the diffusion of inaccurate information under the guidance of initializing-then-refining imputation criterion.
2) Taking the results of Recall@10/NDCG@10 on Amac and Amap for example, RITR with 70% incomplete attributes can still achieve better performance than SAT with 10% ones. These results illustrate that RITR can still achieve high-quality data imputation and feature completion with limited observed signals. Overall, all the above results solidly demonstrate the superiority and robustness of RITR.

4) Hyperparameter Analysis: As seen in (13), RITR introduces two hyperparameters to balance the importance of

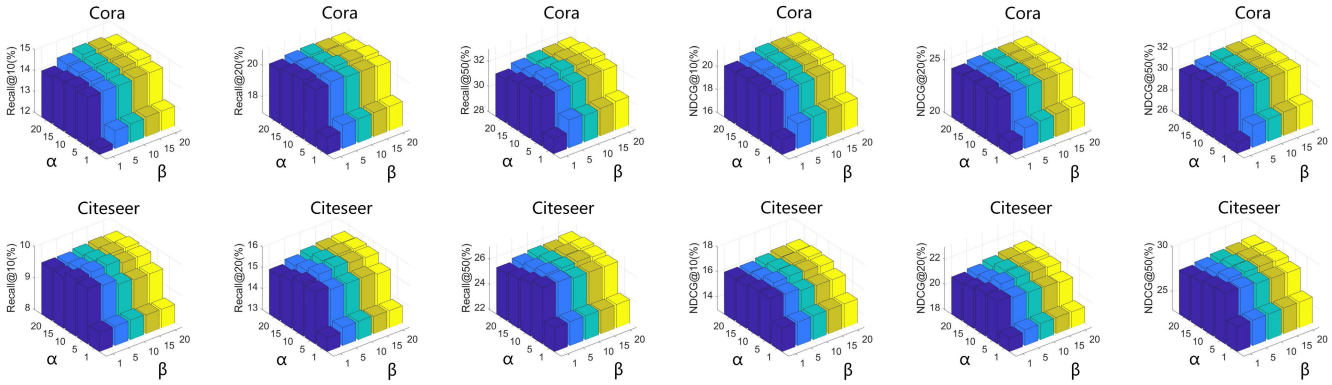


Fig. 5. Sensitivity analysis of RITR with the variation in two hyperparameters. Both the attribute-incomplete and attribute-missing ratios are set to 60%.

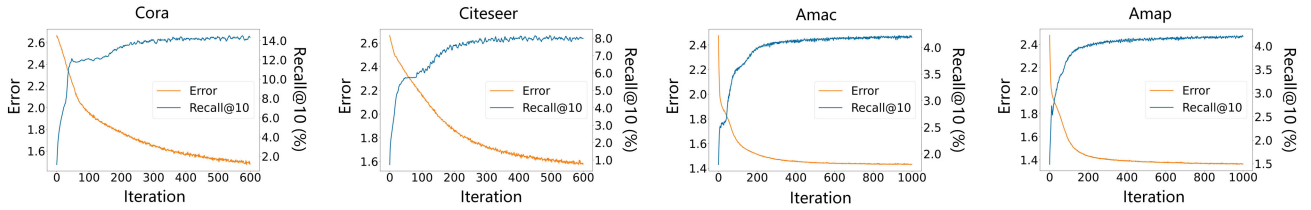


Fig. 6. Illustration of method convergence and performance variation in RITR. X-axis, left Y-axis, and right Y-axis refer to the iteration number, the final objective error, and the Recall@10 performance, respectively. Both the attribute-incomplete and attribute-missing ratios are set to 60%.

different objectives. To show their influence in-depth, we conduct an experiment to investigate the effect of α and β. Note that we first set one to a certain value and then tune the other carefully. Fig. 5 reports the Recall and NDCG performance variation in RITR on Cora and Citeseer when α and β vary from 1 to 20 with a step size of 5. From these subfigures, we can observe that the following conditions hold.

1) Tuning both α and β would cause performance variation and the model performance is more stable in the range of [5, 15], suggesting that searching α and β values from a reasonable hyperparameter region could benefit the model performance.
2) For a certain α value, the performance shows a trend of first rising and then dropping slightly with the variation in β. This indicates that RITR needs a proper coefficient to guarantee the structure-attribute consistency for improving the quality of feature completion. As shown, the performance of the model with a certain β value has similar trends when we change the α value.
3) RITR tends to perform well by setting α and β to 10 according to the results of all the datasets.

5) Convergence and Performance Variation: To illustrate the convergence of the proposed RITR, we record the profiling performance reflected by the Recall@10 metric and plot the objective error of RITR with iterations on four datasets. From these subfigures illustrated in Fig. 6, we can observe that: 1) the Recall@10 metric of RITR first gradually increases to a plateau with an obvious tendency and then keeps stable with a wide range of iterations and 2) RITR can converge within 1000 epochs on four datasets. These results clearly verify the good convergence property of our proposed method and reveal the effectiveness of the learning procedure.

TABLE VIII

PERFORMANCE COMPARISON ON LARGE-SCALE DATASETS. "AM" REFERS TO THE ATTRIBUTE-MISSING SCENARIO. "HA" REFERS TO THE HYBRID-ABSENT SCENARIO. THE BOLDFACE AND UNDERLINE VALUES INDICATE THE BEST AND THE SUBOPTIMAL RESULTS (%), RESPECTIVELY

Type	Dataset	X		X+A	
		Ogbn-A	Ogbn-Ps	Ogbn-A	Ogbn-Ps
AM	GCN	16.09	27.33	29.70	27.38
	GraphSAGE	22.01	27.35	41.15	27.72
	GAT	17.55	27.33	29.70	27.44
	SAT	22.44	27.61	38.83	42.36
	SVGA	21.59	27.30	33.63	36.56
	ITR	<u>24.72</u>	<u>28.12</u>	<u>43.60</u>	<u>49.22</u>
	RITR	**25.41**	**28.90**	**45.75**	**52.37**
HA	GCN	15.89	26.67	18.43	27.37
	GraphSAGE	17.59	26.33	30.62	27.39
	GAT	16.09	26.57	18.41	27.39
	SAT	22.42	26.66	39.62	32.54
	SVGA	20.80	26.90	36.32	34.96
	ITR	<u>24.61</u>	<u>27.61</u>	<u>43.10</u>	<u>45.81</u>
	RITR	**25.33**	**27.93**	**44.27**	**46.46**

D. Performance Comparison on Larger Graphs

To evaluate the performance of RITR on larger graphs, we compare it with six methods on two large-scale datasets, i.e., Ogbn-A (with 169,343 nodes and 1,166,243 edges) and Ogbn-Ps (with 2,449,029 nodes and 61,859,140 edges). Both the attribute-incomplete and attribute-missing ratios are set to 60%. For a fair comparison, all the compared methods are evaluated on the same device and under identical configuration settings. To make the data preprocessing fit into the CPU memory, we adopt the subgraph sampling strategy to pretrain the model. As seen in Table VIII, the proposed ITR and RITR consistently outperform three classical GNNs and other advanced attribute-missing graph machine learning

methods on these larger datasets, once again demonstrating the effectiveness and superiority of the proposed methods.

V. Conclusion and Future Work

Hybrid-absent graphs are ubiquitous in practical applications. However, the corresponding learning problem that significantly influences the performance of the existing graph machine learning methods is still left underexplored. We first propose ITR specifically designed for the attribute-missing scenario. ITR can facilitate the effective negotiation between attribute and structure information, thereby enabling accurate reconstruction of missing values. We further improve ITR and design a variant called RITR to handle hybrid-absent graphs, which can effectively leverage the intimate structure-attribute relationship to guide the imputation of incomplete attributes and use the most trustworthy visible information to implement the missing attribute completion. Extensive experiments on six benchmark datasets have been conducted to compare two proposed methods with the state-of-the-art competitors. These results have solidly demonstrated the superiority and robustness of ITR and RITR on both the profiling and node classification tasks. However, there are still some limitations in the existing attribute-missing or hybrid-absent graph machine learning methods that have not been thoroughly addressed. For instance, the time complexities of most methods are $\mathcal{O}(N^2)$, making them hard to be deployed to various large-scale graph-oriented applications. Future work may extend the proposed RITR to a scalable version with linear scalability [i.e., $\mathcal{O}(BNd)$] via a mini-batch design. Moreover, in the current version, RITR conducts the structure-attribute information interaction and imputation via a simple concatenation. In the future, how to develop a more mathematical hybrid-absent graph machine learning approach to theoretically explain the structure-attribute relationship is another interesting direction.

References

[1] F. M. Bianchi, D. Grattarola, L. Livi, and C. Alippi, "Hierarchical representation learning in graph neural networks with node decimation pooling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 5, pp. 2195–2207, May 2022.

[2] J. Chen et al., "Adversarial caching training: Unsupervised inductive network representation learning on large-scale graphs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7079–7090, Dec. 2022.

[3] H. Huang, Y. Song, Y. Wu, J. Shi, X. Xie, and H. Jin, "Multitask representation learning with multiview graph convolutional networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 3, pp. 983–995, Mar. 2022.

[4] J. Liu, F. Xia, X. Feng, J. Ren, and H. Liu, "Deep graph learning for anomalous citation detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2543–2557, Jun. 2022.

[5] C. Chen, K. Li, W. Wei, J. T. Zhou, and Z. Zeng, "Hierarchical graph neural networks for few-shot learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 1, pp. 240–252, Jan. 2022.

[6] K. Liang et al., "Knowledge graph contrastive learning based on relation-symmetrical structure," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 1, pp. 226–238, Jan. 2024.

[7] K. Liang et al., "Learn from relational correlations and periodic events for temporal knowledge graph reasoning," in *Proc. 46th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2023, pp. 1559–1568.

[8] R. van den Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," 2017, *arXiv:1706.02263*.

[9] M. Zhang and Y. Chen, "Inductive matrix completion based on graph neural networks," in *Proc. Int. Conf. Learn. Represent.*, Apr. 2020, pp. 1–13.

[10] I. Spinelli, S. Scardapane, and A. Uncini, "Missing data imputation with adversarially-trained graph convolutional networks," *Neural Netw.*, vol. 129, pp. 249–260, Sep. 2020.

[11] H. Taguchi, X. Liu, and T. Murata, "Graph convolutional networks for graphs containing missing features," *Future Gener. Comput. Syst.*, vol. 117, pp. 155–168, Apr. 2021.

[12] W. Shen et al., "Inductive matrix completion using graph autoencoder," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 1609–1618.

[13] X. Xie et al., "From discrimination to generation: Knowledge graph completion with generative transformer," in *Proc. Companion Proc. Web Conf.*, Apr. 2022, pp. 162–165.

[14] X. Chen, S. Chen, J. Yao, H. Zheng, Y. Zhang, and I. W. Tsang, "Learning on attribute-missing graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 740–757, Feb. 2022.

[15] W. Tu et al., "Initializing then refining: A simple graph attribute imputation network," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 3494–3500.

[16] B. Perozzi, R. AL-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, p. 701.

[17] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 855–864.

[18] Y. Yang et al., "Self-supervised heterogeneous graph pre-training based on structural clustering," in *Proc. Conf. Neural Inf. Process. Syst.*, vol. 35, Dec. 2022, pp. 16962-16974.

[19] Y. Yang, Z. Guan, J. Li, W. Zhao, J. Cui, and Q. Wang, "Interpretable and efficient heterogeneous graph convolutional network," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1637–1650, Feb. 2023.

[20] Y. Yang, Z. Guan, W. Zhao, W. Lu, and B. Zong, "Graph substructure assembling network with soft sequence and context attention," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 4894–4907, May 2023.

[21] T. N. Kipf and M. Welling, "Variational graph auto-encoders," 2016, *arXiv:1611.07308*.

[22] D. Bo, X. Wang, C. Shi, M. Zhu, E. Lu, and P. Cui, "Structural deep clustering network," in *Proc. Web Conf.*, Apr. 2020, pp. 1400–1410.

[23] W. Tu et al., "Deep fusion clustering network," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 11, May 2021, pp. 9978–9987.

[24] Y. Liu et al., "Deep graph clustering via dual correlation reduction," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2022, pp. 7603–7611.

[25] W. Tu et al., "RARE: Robust masked graph autoencoder," *IEEE Trans. Knowl. Data Eng.*, pp. 1–14, Nov. 2023.

[26] Z. Tao, H. Liu, J. Li, Z. Wang, and Y. Fu, "Adversarial graph embedding for ensemble clustering," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3562–3568.

[27] S. Pan, R. Hu, S.-F. Fung, G. Long, J. Jiang, and C. Zhang, "Learning graph embedding with adversarial training methods," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2475–2487, Jun. 2020.

[28] P. Velickovic, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, "Deep graph infomax," in *Proc. Int. Conf. Learn. Represent.*, Sep. 2019, pp. 1–21.

[29] K. Hassani and A. H. K. Ahmadi, "Contrastive multi-view representation learning on graphs," in *Proc. Int. Conf. Mach. Learn.*, Nov. 2020, pp. 4116–4126.

[30] Z. Peng et al., "Graph representation learning via graphical mutual information maximization," in *Proc. Web Conf.*, Apr. 2020, pp. 259–270.

[31] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, "Graph contrastive learning with adaptive augmentation," in *Proc. Web Conf.*, Apr. 2021, pp. 2069–2080.

[32] L. Gong, S. Zhou, W. Tu, and X. Liu, "Attributed graph clustering with dual redundancy reduction," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 3015–3021.

[33] W. Tu et al., "Attribute-missing graph clustering network," in *Proc. AAAI Conf. Artif. Intell.*, 2024.

[34] G. Ian et al., "Generative adversarial nets," in *Proc. Conf. Neural Inf. Process. Syst.*, Dec. 2014, pp. 1–9.

[35] C. M. Bishop, "Pattern recognition and machine learning," *Stat. Sci.*, 2006.

[36] C. Gao et al., "Neural multi-task recommendation from multi-behavior data," in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Apr. 2019, pp. 1554–1557.

[37] J. You, X. Ma, D. Y. Ding, M. J. Kochenderfer, and J. Leskovec, "Handling missing data with graph representation learning," in *Proc. Conf. Neural Inf. Process. Syst.*, 2020.

[38] C. Huo, D. Jin, Y. Li, D. He, Y.-B. Yang, and L. Wu, "T2-GNN: Graph neural networks for graphs with incomplete features and structure via teacher–student distillation," in *Proc. AAAI Conf. Artif. Intell.*, Dec. 2023, pp. 4339–4346.
[39] D. Jin, C. Huo, C. Liang, and L. Yang, "Heterogeneous graph neural network via attribute completion," in *Proc. Web Conf.*, Apr. 2021, pp. 391–400.
[40] D. He et al., "Analyzing heterogeneous networks with missing attributes by unsupervised contrastive learning," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–13, Mar. 2022.
[41] E. Rossi, H. Kenlay, M. I. Gorinova, B. P. Chamberlain, X. Dong, and M. M. Bronstein, "On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features," in *Proc. Learn. Graphs Conf.*, Dec. 2022, pp. 1–20.
[42] H. Cui, Z. Lu, P. Li, and C. Yang, "On positional and structural node features for graph neural networks on non-attributed graphs," in *Proc. 31st ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2022, pp. 3898–3902.
[43] J. Yoo, H. Jeon, J. Jung, and U. Kang, "Accurate node feature estimation with structured variational graph autoencoder," in *Proc. 28th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2022, pp. 2336–2346.
[44] D. Jin et al., "Amer: A new attribute-missing network embedding approach," *IEEE Trans. Cybern.*, vol. 53, no. 7, pp. 4306–4319, Jul. 2023.
[45] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.-(ICML)*, Jul. 2008, pp. 1096–1103.
[46] H. Gao and H. Huang, "Deep attributed network embedding," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3364–3370.
[47] Ö. Simsek and D. D. Jensen, "Navigating networks by using homophily and degree," in *Proc. Nat. Acad. Sci. USA Amer.*, vol. 105, no. 35, Sep. 2008, pp. 12758–12762.
[48] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Represent.*, Dec. 2014, pp. 1–14.
[49] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, Sep. 2017, pp. 1–13.
[50] W. L. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Conf. Neural Inf. Process. Syst.*, Jun. 2017, pp. 1024–1034.
[51] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, Oct. 2018, pp. 1–12.
[52] X. Huang, Q. Song, Y. Li, and X. Hu, "Graph recurrent networks with attributed random walks," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 732–740.
[53] L. Chen, S. Gong, J. Bruna, and M. M. Bronstein, "Attributed random walk as matrix factorization," in *Proc. Conf. Neural Inf. Process. Syst. Workshop*, Jan. 2019, pp. 1–7.
[54] L. Hu, S. Jian, L. Cao, Z. Gu, Q. Chen, and A. Amirbekyan, "HERS: Modeling influential contexts with heterogeneous relations for sparse and cold-start recommendation," in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2019, pp. 3830–3837.
[55] X. Peng, J. Cheng, X. Tang, J. Liu, and J. Wu, "Dual contrastive learning network for graph clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–11, Apr. 2023.
[56] W. Tu, S. Zhou, X. Liu, C. Ge, Z. Cai, and Y. Liu, "Hierarchically contrastive hard sample mining for graph self-supervised pretraining," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–14, Aug. 2023.

Wenxuan Tu is currently pursuing the Ph.D. degree with the School of Computer, National University of Defense Technology (NUDT), Changsha, China. He has published several papers in highly regarded journals and conferences, such as IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, AAAI, IJCAI, CVPR, NeurIPS, ICML, ACM MM, and SIGIR. His research interests include clustering analysis, graph machine learning, and image semantic segmentation.

Bin Xiao received the B.S. and M.S. degrees in electrical engineering from Shanxi Normal University, Xi'an, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science from Xidian University, Xi'an, in 2012.

He is currently a Professor with the Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include image processing and pattern recognition.

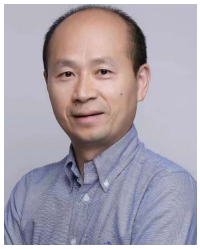
Xinwang Liu (Senior Member, IEEE) received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013.

He is currently a Full Professor with the School of Computer, NUDT. He has published more than 100 peer-reviewed papers, including those in highly regarded journals and conferences such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (IEEE TNNLS), IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, ICML, NeurIPS, ICCV, CVPR, AAAI, and IJCAI. His current research interests include kernel learning and unsupervised feature learning.

Dr. Liu serves as an Associate Editor for the *Information Fusion*, IEEE TRANSACTIONS ON CYBERNETICS, and IEEE TNNLS. More information can be found at https://xinwangliu.github.io.

Sihang Zhou received the bachelor's degree in information and computing science and the M.S. degree in computer science from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2012 and 2014, respectively, and the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2019.

He is currently an Associate Professor with the School of Intelligence Science and Technology, NUDT. His current research interests include machine learning, knowledge graph, and medical image analysis.



Zhiping Cai (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science and technology from the National University of Defense Technology (NUDT), Changsha, China, in 1996, 2002, and 2005, respectively.

He is currently a Full Professor with the School of Computer, NUDT. His current research interests include artificial intelligence, network security, and big data.

Prof. Cai is a Senior Member of China Computer Federation (CCF).

Jieren Cheng received the Ph.D. degree from the School of Computer, National University of Defense Technology, Changsha, China, in 2010.

He is currently a Professor and a Ph.D. Supervisor with Hainan University, Haikou, China. His research interests include cloud computing, artificial intelligence, network security, and intelligent transportation.

Dr. Cheng is a member of China Computer Federation (CCF).