Reconstructing the Unseen: GRIOT for Attributed Graph Imputation with Optimal Transport

Richard Serrano, Charlotte Laclau, Baptiste Jeudy, Christine Largeron

Reconstructing the Unseen: GRIOT for Attributed Graph Imputation with Optimal Transport

Richard Serrano[1] [0009−0009−4946−896X] ✉,
Charlotte Laclau[2] [0000−0002−7389−3191],
Baptiste Jeudy[1] [0009−0000−8126−2608], and
Christine Largeron[1] [0000−0003−1059−4095]

[1] Laboratoire Hubert Curien, Saint-Étienne 42000, France
{first-name}.{last-name}@univ-st-etienne.fr
[2] Télécom Paris, Institut Polytechnique de Paris, France
{first-name}.{last-name}@telecom-paris.fr

Abstract. In recent years, there has been a significant surge in machine learning techniques, particularly in the domain of deep learning, tailored for handling attributed graphs. Nevertheless, to work, these methods assume that the attribute values are fully known, which is not realistic in numerous real-world applications. This paper explores the potential of Optimal Transport (OT) to impute missing attribute values on graphs. To proceed, we design a novel multi-view OT loss function that can encompass both node feature data and the underlying topological structure of the graph by utilizing multiple graph representations. We then utilize this novel loss to train efficiently a Graph Convolutional Neural Network (GCN) architecture capable of imputing all missing values over the graph at once. We evaluate the interest of our approach with experiments both on synthetic data and real-world graphs, including different missingness mechanisms and a wide range of missing data. These experiments demonstrate that our method is competitive with the state-of-the-art in all cases and of particular interest on weakly homophilic graphs.

Keywords: Attributed Graph · Missing Data Imputation · Optimal Transport

1 Introduction

Graphs have become an indispensable tool for modeling and solving a wide range of practical problems. From transportation networks to protein-protein interactions, graphs provide a natural and versatile representation framework for modeling relationships of various kinds. In this context, so-called attributed graphs possess a valuable set of information about the objects whose relationships they model, e.g., the personal information of users for online social networks, or the properties of atoms for a molecule. However, for real-world applications, attributed graphs often suffer from missing data [17]. For instance, it is

quite common for traditional online social networks, usually modeled as graphs of users, to have missing attribute values, as some of the information filled in by users is not mandatory (e.g., gender, age). This will result in missing node attributes in a graph.

Imputing missing data is a long-standing challenge in statistics that has been at the forefront of research and practice for decades [19]. This problem arises when datasets contain missing or incomplete information, which can lead to biased analyses, reduced statistical power, and inaccurate results.

Missing data may be the result of different factors, also referred to in the literature as *mechanisms* [25]. The simplest case is MCAR (Missing Completely At Random), where the missingness of the attribute values (true or false) can be modeled by i.i.d. random variables. For instance, a social platform's localization data might be lost due to occasional technical glitches. In the MAR (Missing At Random) case, the missing data probability depends on other observed attributes. For example, in social media profiles, MAR might occur if the probability of users omitting their hobbies depends on their gender. MNAR (Missing Not At Random) is the most challenging case, where missing data probability depends on unobserved variables. For instance, users may choose not to share their income due to personal factors, unaccounted for in the dataset. Depending on the mechanism, imputation can be more or less difficult.

In graph data, one approach to address missing node attributes is to cast the problem as imputing missing information in tabular data, given that node attributes are structured as a matrix. However, this method does not use the crucial structural dependencies inherent in graph data. Therefore, addressing missing information on graphs requires methods that respect and leverage the underlying graph structure [10].

Most imputation methods on graphs assume homophily [24], but as it turns out, real graphs are sometimes heterophilic [34]. A graph is homophilic if nodes sharing similar attributes tend to be more often connected than those with different attributes: "birds of a feather flock together" [20]. In this paper, we are interested in the problem of the imputation of missing node attributes, whatever the nature of the graph, homophilic or heterophilic, or the missingness mechanism (MCAR or MNAR). To proceed, we propose to exploit the Optimal Transport (OT) theory to impute missing node attributes on graphs. In recent years, OT has proved remarkably successful in machine learning notably for missing values imputation on tabular data [23] and more recently with applications on graphs including node embedding [33], fair edge prediction [18], graph prediction [2], to name a few. The intuition of using OT for imputation [23] is that the distance between two random samples from the data distribution should be small (using Wasserstein distance from optimal transport theory). Thus, a good imputation of missing values should minimize this distance between many pairs of random samples. This goal is particularly suitable for OT-based distances, as they can be (and have already been) used in gradient-based optimization as valid loss terms due to their attractive differentiability properties [21,2]. However, the classic Wasserstein distance does not take the graph topology into account.

Contributions. We propose to use a novel distance, *multi-view Wasserstein* (MultiW), able to take into account any representation of the graph, such as attributes, topology, but also hierarchy, spectral decomposition, and more. This distance is also more flexible and more computationally efficient than similar multi-view loss functions, such as FGW [29] or OTT [13]. The MultiW distance is used as a loss to train a Graph Neural Network (GNN) model able to impute missing attributes on a graph. The main distinctive feature of our approach is its ability to use the trained imputer on new nodes without the need to be retrained (contrary to state-of-the-art FP [24]). This feature is of particular interest as many graphs, such as social networks, are inherently dynamic.

Summing up, our contributions embed (1) the creation of an efficient Multi-view loss function referred to as MultiW; (2) a framework for graph missing attributes imputation `GRIOT` (**GR**aph **I**mputation with **O**ptimal **T**ransport) ; an extensive empirical study of the performance of our approach and the most recent state-of-the-art methods on a very wide variety of scenarios, whereas most studies generally focus on a very small subset of these scenarios. The code is available online*.

2 Related Works

Over the years, researchers have developed a wide range of techniques to tackle the problem of missing data imputation [28,26,32]. Despite the progress made, the field of research for data imputation remains dynamic due to evolving data complexity. In this paper, we specifically address missing node attribute values in data structured as graphs. There has been a renewal of interest in graphs, driven particularly by the rise of Graph Neural Networks (GNNs) that require complete attributes.

While various approaches like SAT[3], GCNMF[27], and PaGNN[11] have aimed to adapt GNNs to this context, they primarily emphasize task performance rather than imputation quality. Conversely, some methods employ GNNs for graph completion [22,1], focusing on attribute matrix reconstruction over task performance, but they often encounter scalability challenges.

Finally, to the best of our knowledge, Feature Propagation (FP) [24] is currently the state-of-the-art method for missing node attribute imputation. FP is a diffusion-based attribute reconstruction approach that allows imputation on the graphs upstream to the node classification task. As such, it is not tied to any particular model or architecture for solving the final task. However, similarly to the aforementioned approaches, FP assumes a strong attribute-based homophily in the graph to impute the missing attributes. This means that nodes in the graph are more likely to be connected to other nodes that have similar attributes.

Our approach is at the crossroads of these methods. Much like FP, we impute the missing node attributes matrix in an initial pre-processing step. However,

* Code available at: github.com/RichardSrn/GRIOT

our approach involves training an imputer that relies on GNNs. Our framework is also mainly different in the design of the loss function, which simultaneously takes into account graph topology and node attributes, potentially assigning different weights to each. This distinctive property makes our approach well-suited to graphs with low homophily, and for complex missing data mechanisms, in contrast with SOTA methods, which have not been evaluated in this context before, and which are outperformed by `GRIOT` according to our experiments.

3 Multi-view Optimal Transport Loss for Attribute Imputation

Hereafter, we define our notations and provide a reminder about Optimal Transport (OT) and the Wasserstein distance. We then propose a novel loss function that extends OT to enable graph attribute imputation.

3.1 Notations

We denote by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, F)$, an undirected and attributed graph, where $\mathcal{V} = (v_i)_{1 \leq i \leq n}$ is the set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ the set of edges represented by an adjacency matrix $A = (a_{ij})_{i,j=1}^{n} \in \{0,1\}^{n \times n}$, such that $a_{ij} = 1$ if $(v_i, v_j) \in \mathcal{E}$, 0 otherwise and $F = (f_i)_{i \leq n} \in \mathbb{F}^{n \times d}$ are the node attribute vectors. In this paper, we consider real or binary attributes: $\mathbb{F} = \mathbb{R}$ or $\mathbb{F} = \{0,1\}$. In the context of missing data, some values in F are not observable. These missing values are encoded as 0 in a binary mask $\Omega \in \{0,1\}^{n \times d}$. The ground truth values are denoted F^{gt} and $F = F^{gt} \odot \Omega + \texttt{NaN} \odot (1 - \Omega)$, where $\odot$ is the Hadamard product and `NaN` denotes missing values. The missing value imputation problem is to recover an approximation $\hat{F}$ of F^{gt} from $\mathcal{G}$ and Ω. In the following, a *view* ζ_j of a graph $\mathcal{G}$ is a collection of n vectors in a z_i dimensional space. Formally, ζ_i is a $n \times z_i$ matrix, with its ℓ-th vector associated to the ℓ-th node of $\mathcal{G}$ (the nodes attribute matrix F is such a view).

There exist multiple tasks associated with graph analysis including node classification, edge prediction, and community detection to name a few. In this paper, we consider node classification as our auxiliary task; missing data imputation being the primary task. Therefore, we assume a label associated with each node: $\mathcal{C} : \mathcal{V} \to \{1, \cdots, k\}$. The goal of node classification is to learn a classification function that maps each node v_i in the graph to a class label, i.e., we aim to learn $\epsilon : \mathcal{V} \to \{1, \cdots, k\}$ s.t. $\epsilon(v_i) = C(v_i), \forall i \in \{1, \cdots, n\}$. Note that because our imputation procedure is done before solving the task, one can easily apply our method to any subsequent task.

3.2 Optimal Transport and Wasserstein Distance

We present theoretical notions from Optimal Transport (OT) [30] restricted to two weighted sets of data points in $\mathbb{R}^d$, useful for understanding the sequel.

Given two weighted sets of data points (X, w_1) and (Y, w_2) where $X = \{x_i\}_{i \leq n_1} \in \mathbb{R}^{d \times n_1}$ and $Y = \{y_j\}_{j \leq n_2} \in \mathbb{R}^{d \times n_2}$ are composed respectively of n_1 and n_2 vectors of size d. The weight vector w_1 of size n_1 (resp. w_2 of size n_2) defines the weights of each vector of X (resp. Y). Each weight vector is a discrete probability distribution on X (resp. Y), meaning that all the weights are positive, and they sum to one.

The goal of OT is to find a transport plan of minimal cost between (X, w_1) and (Y, w_2). This minimal cost is called the Wasserstein distance between (X, w_1) and (Y, w_2). A transport plan is represented by a matrix π such that $\pi_{i,j}$ is the weight transported from data point x_i to data point y_j. The constraints are that the total weight transported from x_i must sum to $w_1(i)$ and the total weight transported to y_j must sum to $w_2(j)$. More formally, π must belong to the set of valid transportation plans $\Pi(w_1, w_2)$:

$$\Pi(w_1, w_2) = \{\pi \in \mathbb{R}_+^{n_1 \times n_2} \mid \pi \mathbb{1}_{n_2} = w_1,\ \pi^T \mathbb{1}_{n_1} = w_2\}, \tag{1}$$

where $\mathbb{1}_n$ is the vector $(1, 1, ..., 1)^T$ of dimension n, and $\cdot^T$ the transposition operation.

The costs $m_{i,j}$ of transporting one unit of weight from x_i to y_j are given in a matrix $M^{X,Y} = (m_{i,j})_{1 \leq i \leq n_1, 1 \leq j \leq n_2}$. This matrix is usually computed as the ℓ_2 norm between the points of X and Y.

Finally, the optimal plan is found by solving the following regularized minimization problem:

$$\pi((X, w_1), (Y, w_2)) = \underset{\pi \in \Pi(w_1, w_2)}{\operatorname{argmin}} \langle M^{X,Y}, \pi \rangle_F + \varepsilon H(\pi) \tag{2}$$

where $\langle \cdot, \cdot \rangle_F$ is the Frobenius inner product, ε is the regularization hyperparameter, and H is the entropy.

The Wasserstein distance [12] is the cost associated with the optimal plan:

$$\mathcal{W}((X, w_1), (Y, w_2)) = \min_{\pi \in \Pi(w_1, w_2)} \langle M^{X,Y}, \pi \rangle_F + \varepsilon H(\pi) \tag{3}$$

Being equipped with all the necessary material from OT theory, we can now move on to the description of MultiW, our newly introduced loss function.

3.3 Definition of the MultiW Loss Function

Graphs are complex objects, that can be represented through various means, each offering distinct insights. For example, an adjacency matrix captures pairwise node connections, reflecting first-order proximity. On the other hand, the Laplacian matrix [4] conveys information on node degrees and their relationships, providing a deeper understanding of the graph's structure beyond pairwise connections. Finally, the node attribute matrix is yet another way to represent the nodes of the graph.

To capture all these properties at once, we aim to design a loss function that can simultaneously leverage these diverse views of a graph to impute missing

data in the feature matrix. Note that while our focus is on missing attributes, our approach could be extended to the imputation of missing information on any other view, notably the adjacency matrix for link completion.

Motivation for OT. We opt for OT theory as it provides a clever way of estimating the discrepancy between distributions. We assume that the distribution of imputed values should resemble that of the known values, especially when taking into account multiple views of the graph, i.e. the optimal transport distance between these distributions should be small.

General Definition. Let us consider a graph $G = (\mathcal{E}, \mathcal{V})$ with n nodes and $\zeta = (\zeta_i)_{i \leq q}$ be q views representing G in different spaces, such that for all i, ζ_i is a $n \times z_i$ matrix. Let $\alpha = (\alpha_i)_{1 \leq i \leq q} \in [0,1]^q$ be the views' weights such that $\sum_{i=1}^{q} \alpha_i = 1$. To proceed, we quantify the distance between random subgraphs to evaluate the gap between the distributions. Therefore, we consider two subgraphs of $\mathcal{G}$ respectively obtained by randomly selecting two subsets of nodes $\mathcal{V}^1$ and $\mathcal{V}^2$ from $\mathcal{V}$, and their respective views $(\zeta_i^1)_{1 \leq i \leq q}$ and $(\zeta_i^2)_{1 \leq i \leq q}$.

To compute an optimal transport between $\mathcal{V}^1$ and $\mathcal{V}^2$ with the q views of ζ, one could solve the OT problem q times independently, and get q different transport plans. In our case, we are interested in solving the OT problem in a way that takes into account the q views simultaneously, resulting in a single transportation plan.

With MultiW, we propose to solve a unique optimization problem considering all views at once:

$$\text{MultiW}_\alpha((\zeta_i^1)_{i \leq q}, (\zeta_i^2)_{i \leq q}) = \min_{\pi \in \Pi(w_1, w_2)} \sum_{i=1}^{q} \alpha_i \langle M^{\zeta_i^1, \zeta_i^2}, \pi \rangle_F + \varepsilon H(\pi) \quad (4)$$

However, this issue is not solvable in a reasonable time with existing tools, such as the `POT` [9] library, because of our q optimization objectives. Nonetheless, as a direct consequence of the linearity of the Frobenius inner product, we have:

$$\sum_{i=1}^{q} \alpha_i \langle M^{\zeta_i^1, \zeta_i^2}, \pi \rangle_F = \langle \sum_{i=1}^{q} \alpha_i M^{\zeta_i^1, \zeta_i^2}, \pi \rangle_F \quad (5)$$

Hence, using the linearity, the q optimization problems, defined in equation (4), can be simplified to one as defined in equation (6):

Definition 1. *Multi-view Wasserstein. Given a graph $\mathcal{G}$, and q views $\zeta = (\zeta_i)_{i \leq q}$. For any two subgraphs corresponding to the subsets of nodes $\mathcal{V}_1$ and $\mathcal{V}_2$, and their respective views $(\zeta_i^1)_{1 \leq i \leq q}$ and $(\zeta_i^2)_{1 \leq i \leq q}$; given the weights of nodes of the subgraphs w_1 and w_2; given $\alpha = (\alpha_i)_{1 \leq i \leq q} \in [0,1]^q$ such that $\sum_{i=1}^{q} \alpha_i = 1$, let $M^{\zeta_i^1, \zeta_i^2}$ be the cost matrix between the subgraphs, according to the i-th view, then:*

$$\text{MultiW}_\alpha((\zeta_i^1)_{i \leq q}, (\zeta_i^2)_{i \leq q}) = \min_{\pi \in \Pi(w_1, w_2)} \langle \sum_{i=1}^{q} \alpha_i M^{\zeta_i^1, \zeta_i^2}, \pi \rangle_F + \varepsilon H(\pi) \quad (6)$$

Remark: The weights w_1 (resp. w_2) can be set to a uniform distribution $\forall i \in [\![1, n_1]\!]$, $(w_1)_i = \frac{1}{n_1}$ or proportional to the nodes' degree.

The optimization problem presented in equation (6) can be solved by the Sinkhorn-Knopp's fixed point iteration algorithm [6], and the solution is a well-defined transport plan.

3.4 Instantiation of MultiW Loss with Attributes and Structure

In this work, our focus lies on imputing missing node attributes in a graph. Therefore, we use equation (6) to integrate two views. These two views encapsulate both the structural characteristics and the node attributes matrix F associated with the graph. To represent the structural aspects, a straightforward representation is through the adjacency matrix A. To effectively incorporate this representation into our loss function, we propose the computation of a proximity matrix $P \in \mathbb{N}^{n \times n}$ which is defined as $P = (p_{i,j})$ where $p_{i,j}$ is the geodesic path length between the nodes v_i and v_j. Now, based on the two views P and F, we can leverage the MultiW loss to estimate the mean distance between random subgraphs of $\mathcal{G}$. We operate under the assumption that nodes with comparable roles in the graph exhibit similar attribute distributions, a well-imputed graph thus yielding a smaller MultiW distance.

Let us consider $\mathcal{G}^1$, a subgraph of G, and the corresponding vertices $\mathcal{V}^1$. Let $P_1 \in \mathbb{N}^{|\mathcal{V}^1| \times n}$ be the sub-matrix of P associated to $\mathcal{G}^1$ defined as $P_1 = \{p_{i,j} | v_i \in \mathcal{V}_1, v_j \in \mathcal{V}\}$. Unlike a sub-adjacency-matrix, P_1 is a rectangular matrix, representing the relative position of each node in $\mathcal{V}_1$ to every other node in $\mathcal{V}$ as shown in Figure 1. Let F^1 be the sub-features-matrix associated to $\mathcal{G}^1$. Similarly, P^2, F^2 are views corresponding to a random sub-graph $\mathcal{G}^2$.

The distance between $\mathcal{G}^1$ and $\mathcal{G}^2$ can be computed as:

$$\text{MultiW}_\alpha((F_1, P_1), (F_2, P_2)) = \min_{\pi \in \Pi(w_1, w_2)} \langle M_\alpha, \pi \rangle_F + \varepsilon H(\pi) \tag{7}$$

where $M_\alpha = (1-\alpha)M^{F_1,F_2} + \alpha M^{P_1,P_2}$, and $(1-\alpha)$, α are the weights attributed to, respectively, the features and the structure.

Remark: The Fused-Gromov-Wasserstein (FGW) [29] could also be used to compute the distance between $\mathcal{G}^1$ and $\mathcal{G}^2$. However, FGW transports pairs of nodes on pairs of nodes yielding a $O(n^4)$ complexity of the Frobenius product computation of the loss versus $O(n^2)$ for MultiW. Moreover, FGW only takes into account the edges of the two subgraphs contrary to MultiW which uses the proximity matrix with all other nodes of the whole graph. Finally, the complexity of MultiW grows linearly with the number of views considered, which makes it very efficient when compared to other multi-views OT approaches such as Optimal Tensor Transport (OTT) [13].

4 Imputing Missing Attributes with MultiW Loss

Next, we describe the architecture that we designed to train an imputer that optimizes the `GRIOT` loss with the ability to impute all features in parallel.

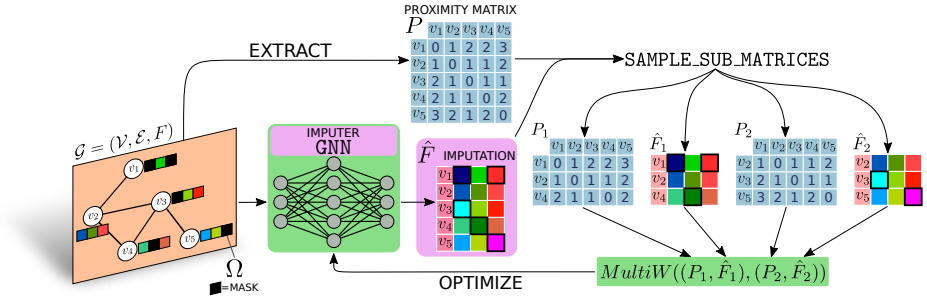

Fig. 1. Architecture of the GRIOT Framework. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with nodes attributes F, and a mask of missing data Ω as input, GRIOT decomposes in 2 main elements: the GCN Imputer and the MultiW Loss Function used to optimize it. The framework outputs the last imputed attributes matrix $\hat{F}$ and the GNN imputer trained and ready to be reused on new nodes with missing attributes.

4.1 Architecture of GRIOT

The overall architecture of GRIOT is presented in Figure 1, a detailed pseudocode is shown in the additional material (see Algorithm GRIOT), and the code is available online*.

Input of GRIOT. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a matrix of attributes F, and a mask Ω of missing features GRIOT aims at reconstructing the node attributes matrix, $\hat{F}$. Our imputer takes as input the adjacency (A) and the feature (F) matrices. However, the missing values of F must be filled in at the first iteration. To this end, we initialize $\hat{F}$ by imputing the missing values with normal random values such that $(\mu_j, \sigma_j)_{j \leq d}$ are the average and standard deviation of each feature over observed ones.

$$\forall i \leq n,\ \forall j \leq d,\ \hat{F}_{i,j} = \begin{cases} F_{i,j}, \text{ if } \omega_{i,j} == 1 \text{ (i.e. } F_{ij} \text{ is observed)} \\ \hat{F}_{i,j} \sim \mathcal{N}(\mu_j, \sigma_j) \text{ otherwise} \end{cases} . \quad (8)$$

The loss takes as input the proximity matrix P computed from the adjacency matrix such that $\forall 1 \leq i \leq n,\ \forall 1 \leq j \leq n$, $p_{i,j}$ is the length of the geodesic path between the nodes v_i and v_j, and the imputed feature matrix $\hat{F}$.

Architecture and Training of the Imputer. The imputer is a Graph Convolutional Network (GCN) [16] that takes as input $\hat{F} \in \mathbb{F}^{n \times d}$ after the initialization of the missing values and A. The output of the GCN is the imputed feature matrix, F^{imp}. Formally, we get the following equations for a GCN with $\ell = 1, \cdots, L$ layers:

$$H_1 = \sigma(\hat{D}^{-1/2}\hat{A}\hat{D}^{-1/2}\hat{F}\Theta_1)$$
$$H_{\ell+1} = \sigma(\hat{D}^{-1/2}\hat{A}\hat{D}^{-1/2}H_\ell\Theta_{\ell+1})$$
$$F^{imp} = \sigma'(H_L\Theta_{L+1}^T + \Theta_{L+1})$$

where $\hat{A} = A + I$ denotes the adjacency matrix with inserted self-loops, $\hat{D}$ is the diagonal degree matrix, $(H_i)_{i \leq L}$ are hidden states, σ and σ' are activation functions, and $(\Theta_i)_{i \leq L+2}$ are the learned parameters.

Finally, the mask of observed features is applied to the imputed matrix such that only missing features are replaced:

$$\hat{F} = F \odot \Omega + F^{imp} \odot (1 - \Omega) \tag{9}$$

where $\odot$ is the element-wise product. The GCN is trained by minimizing the MultiW loss (see definition 1). To compute the loss, we consider two subgraphs $\mathcal{G}^1$ and $\mathcal{G}^2$ from $\mathcal{G}$, defined from two subsets of nodes randomly drawn from $\mathcal{V}$, and as explained in Section 3.4, we use the MultiW loss function to estimate the distance between them. This operation is carried out with n_p couples of random subgraphs, and sum as $loss = \sum_{i=1}^{n_p} \text{MultiW}((F_i^1, P_i^1), (F_i^2, P_i^2))$, before being back-propagated through the imputer. The whole process is repeated `epochs` times, with a new imputation at each epoch.

Output of `GRIOT`. The output of `GRIOT` is the last version of the imputed feature matrix $\hat{F}$, and the imputer itself. The trained imputer offers the possibility to perform inductive feature imputation. Indeed, an important difference between our approach and current SOTA methods is that it goes beyond one-shot imputation on a graph and can impute missing features on new nodes.

4.2 Accelerating the Imputation

The training of machine learning models as imputers has been studied in the literature on tabular data, but these approaches are generally based on the Round-Robin principle of sequentially training one ML model imputer for each feature (not parallelizable as each ML imputer depends on the previous ones). As a result, the Round-Robin principle suffers from scalability and efficiency problems that limit its use for high-dimensional data. Indeed, all real graphs considered in this paper have in the order of magnitude of 10^3 attributes.

In contrast, our approach can impute all features simultaneously using one GNN imputer, providing a parallelizable solution, especially when utilizing GPUs.
Complexity Analysis. The theoretical time complexity analysis of `GRIOT` yields a complexity of $O(d \times m \times n^2 \times \texttt{n\_epochs} \times \texttt{n\_pairs})$, where d is the number of features, m is the number of views, n is the batch size, `n_epochs` and `n_pairs` are parameters of Algorithm `GRIOT` (c.f. additional material).

5 Experimental Analysis

Now, we propose an extensive comparison of `GRIOT` against different SOTA approaches while covering a rich variety of scenarios: two different masking strategies (MCAR and MNAR), different percentages of missing information, and different levels of homophily. In addition, our evaluation unfolds across two dimensions: the quality of the values imputed by the different approaches and their

Table 1. Datasets summary. h_{obs} is the observed homophily, h_{exp} is the expected homophily from a random graph and $h_{ratio} = h_{obs}/h_{exp}$. $+$ (resp. $-$) denotes a strong (resp. weak) homophily level. k is the number of classes.

	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{E}\rvert$	$\mathbb{F}$	k	sparsity	h_{obs}	h_{exp}	h_{ratio}
SBM2$^-$ (low hom.)	796	2407			2.8%	0.13	0.34	0.38
SBM1$^\sim$ (med hom.)	794	1939	$[0,1]^3$	4	3.1%	0.55	0.61	1.11
SBM0$^+$ (high hom.)	770	2085			3.1%	0.90	0.36	2.50
SBM	794	1939	$[0,1]^3$	4	2.5%	0.76	0.35	2.19
CORNELL$^-$	127	159			94.3%	0.13	0.32	0.42
TEXAS$^-$	129	171	$\{0,1\}^{1,702}$	5	95.0%	0.13	0.42	0.31
WISCONSIN$^-$	168	232			93.6%	0.17	0.30	0.55
CITESEER$^+$	2,120	3,679	$\{0,1\}^{3,702}$	6	99.1%	0.74	0.19	3.90
CORA$^+$	2,485	5,069	$\{0,1\}^{1,433}$	7	98.7%	0.80	0.18	4.50
PUBMED$^+$	19,717	44,324	$[0,2]^{499}$	3	89.9%	0.80	0.36	2.20

impact on node classification. Note that, most recent works focus on the latter aspect for homophilic graph and with MCAR masking. Overall, our goal is to answer the following research questions:

1. Can we outperform state-of-the-art methods by using the OT theory in `GRIOT` on graph features imputation, and on node classification?
2. Does the masking strategy impact the imputation?
3. Theoretically, `GRIOT` can impute new nodes without recycling. How does it behave in such a scenario?

In this section, we first describe the experimental protocol setup to address these questions and then present the results obtained for all aforementioned scenarios.

5.1 Experimental Protocol

Baselines. We include both *naive* and SOTA baselines. For the naive baseline, we compare with a K-nearest-neighbors-based principle and simply impute the *average* of the features of nodes directly connected to the node presenting missing values. The strong baselines comprise three models. The first one, `PaGNN` [11], considers the mask Ω when classifying nodes. The second baseline, OT-TAB [23] is also an OT-based imputation approach but for tabular data only leaving out the graph's structure. It corresponds to two algorithms, a one-shot imputation referred to as OT-TAB and a Round-Robin-based version referred to as OT-TAB-rr that, similar to us, allows the training of an imputer. Note that we could not run the OT-TAB-rr algorithm on all datasets for time complexity reasons. The third and strongest baseline is Feature Propagation (FP) [24] which is currently the best approach among SOTA for this task.

Datasets. We evaluate our algorithm and baseline methods on diverse graphs, both synthetic and real, covering various homophily levels as summarized in Table 1. Homophily is defined as the tendency for nodes with similar attributes to be more likely connected [14]. Table 1 presents the expected homophily h_{exp} derived from a randomly drawn graph with equivalent node count and edge probability, alongside the observed homophily h_{obs} in the given graph. The ratio $h_{ratio} = \frac{h_{obs}}{h_{exp}}$ provides a key insight, where a value exceeding 1 signifies high homophily, while a value below 1 indicates weak homophily.

We use the Stochastic Block Model (SBM) to generate synthetic graphs with different homophily levels, spanning from low to medium and high homophily. Each dataset has four imbalanced clusters, and specific parameters for generation can be found in the online code repository*. In addition to the synthetic graphs, our evaluation encompassed three datasets from the WebKnowledge-Base (WebKB) [5] (weakly homophilic, marked with a superscript "$-$" sign) and three citation graphs from the citation network (Planetoid) datasets [31] (highly homophilic, marked with a superscript "$+$" sign).

Missing Data Masking. We use two strategies to build the mask Ω: Missing Completely At Random (MCAR) and Missing Not At Random (MNAR). For MCAR, we draw $\Omega = (\omega_{i,j})_{i \leq n,\ j \leq d}$ i.i.d. such that $\forall (i,j),\ \omega_{i,j} \sim \mathcal{B}(p)$ follows a Bernoulli distribution, where p is the probability of values being set to 0 in the feature matrix. MCAR exhibits no correlation with the data or graph structure. In MNAR, we consider the self-masked context, where $P(\omega_{i,j} = 0)$ depends on the values of the unobserved data itself (extreme values are more likely to be missing) and the characteristics of the graph's structure (nodes with lower degree are more prone to have missing data). Finally, we also vary the level of missing data during experiments, maintaining consistency at 20%, 50%, and 80%.

Evaluation Metrics. Evaluation metrics fall into two categories. The first category assesses the quality of the imputed values by comparing the imputed matrix $\hat{F}$ to ground truth values F^{gt}, using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), where lower is better. The second evaluates imputation's impact on node classification done with a Graph Neural Network after imputation and is based on accuracy and ROC-AUC scores, where higher is better. We use the Mann-Whitney U test with a 5% p-value over 5 runs to determine significant results.

Remark: Results presented in the following are a subset of our experiments. Additional results are available in the additional material.

Imputer architecture. To build the reconstructed feature matrix $\hat{F}$, we employ a Graph Neural Network (GNN). The imputer accepts as input the edge index $\mathcal{E}$ and the imputed features $\hat{F}$, which have dimensions $n \times d$. Consequently, both the input and output of the imputer are of size d. We optimized the imputer's architecture through cross-validation, resulting in a model with 2 layers of Graph Convolutional Network (GCN) and 1 linear layer, with respective dimensions $(d, \lceil\sqrt{d}\rceil)$, $(\lceil\sqrt{d}\rceil, \lceil\sqrt{d}\rceil)$, and $(\lceil\sqrt{d}\rceil, d)$. This architecture enhances graph

abstraction and improves overall reconstruction quality. Furthermore, we introduced a 50% dropout rate to increase the model's robustness against overfitting.

The imputer's parameters are optimized using the Adam optimizer [15], with a learning rate set to 0.01 and a weight decay of 10^{-5}.

Classifier architecture. To evaluate the performance of all imputers on the node classification task, we define a GNN classifier. This classifier takes $(\mathcal{E}, \hat{F})$ as input, where $\hat{F}$ has dimensions $n \times d$, and each node is assigned to one of k distinct classes. The classifier consists of 2 `Cheb` layers [7] and 1 linear layer, with dimensions $(d, \lceil\sqrt{d}\rceil)$, $(\lceil\sqrt{d}\rceil, \lceil\sqrt{d}\rceil)$, and $(\lceil\sqrt{d}\rceil, k)$, respectively.

We determined the type of layers through cross-validation, although we spent a short time optimizing the layers' hyperparameters, as the primary goal was to assess the impact of imputation on classification performance.

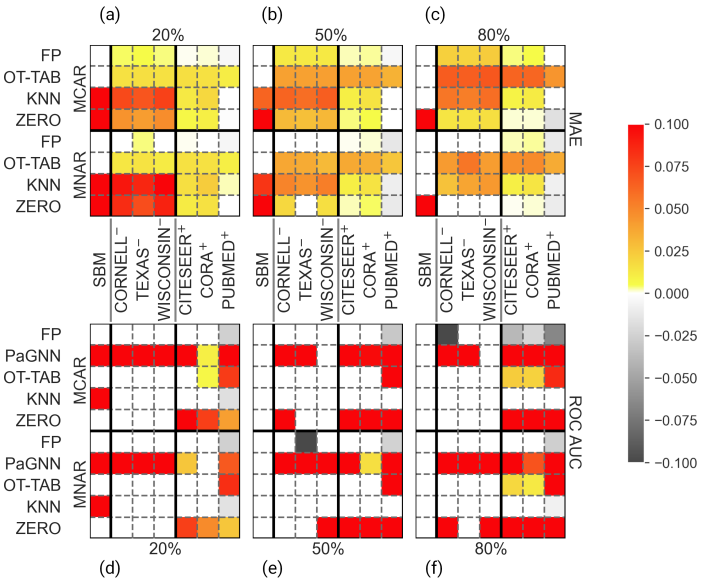

Fig. 2. Comparison of `GRIOT` v.s. baselines. (a,b,c) **imputation** MAE↓, (d,e,f) **classification** ROC AUC↑, on multiple datasets with (1) varying degrees of missing data: (a,d) 20%, (b,e) 50%, and (c,f) 80%, and (2) varying missingness mechanisms: MCAR (upper parts of the matrices) and MNAR (lower parts). In the visualization, white squares denote non-significant differences, grayscale squares indicate instances where `GRIOT` is outperformed, and colored squares represent significant improvements of `GRIOT` over baseline methods, with colors ranging from yellow (small differences) to red (larger differences).

5.2 Imputation Quality v.s. Node Classification Accuracy

We start by assessing the performances of all methods, with a distinction between the accuracy of the imputed values and the impact on node classification accuracy.

Imputed Values. Looking at the quality of the imputed values, Figure 2 (a, b, c) shows the MAE obtained by all baselines, with a distinction between homophilic and heterophilic graphs with different masking strategies, MCAR in the upper parts and MNAR in the lower parts, and percentages of missing attributes varying from 20% (a), 50% (b) to 80% (c). This figure presents pairwise comparisons between `GRIOT` and each of the baselines; colored squares (from yellow to red) correspond to the case when `GRIOT` significantly outperforms other baselines. Overall, `GRIOT` performs much better for this particular task than its competitors, and this difference is even more significant on heterophilic graphs. However, we note that despite the high sparsity of the attributes, the strategy that consists of imputing a zero in place of all missing attributes, indicated by ZERO, performs poorly. Now, taking a closer look at the masking strategy, we can see that the results of KNN and OT-TAB are consistent with the ones obtained when using MCAR or MNAR. Finally, we observe that the differences in FP performances become more pronounced as the percentage of missing data increases.

Node Classification. Figure 2 (d, e, f) presents the AUC score differences between all baselines versus `GRIOT`, it reads the same way as Figure 2 (a, b, c). We observe that, although `GRIOT` showed better imputation performance compared to baselines, this superiority does not systematically translate into a noticeable improvement in the node classification task. Indeed, when dealing with 20% and 50% of missing attributes, we see that `GRIOT` obtains comparable AUC with almost all baselines, including FP on all datasets. We also note that `GRIOT` is always outperforming PaGNN. Now moving on to the 80% of missing attributes, FP is obtaining better results for all the homophilic graphs. This comes as no surprise, as FP was shown to perform extremely well in these extreme types of scenarios [24]. Finally, it is worth remarking that, with 80% of missing attributes, `GRIOT` significantly outperforms OT-TAB on the homophilic graphs.

Table 2. Average of the best α over the type of graph and the tasks: imputation and node classification.

	SBM		WebKB$^-$		Planetoid$^+$	
Masking	MCAR	MNAR	MCAR	MNAR	MCAR	MNAR
Imput.	0.50	0.42	0.31	0.50	0.14	0.33
Classif.	0.42	0.42	0.25	0.31	0.33	0.58

Are both objectives always aligned? To get more insights into the correlation between imputation quality and node classification, we take a closer look at the hyper-parameter α of `GRIOT`, which determines the importance of each view. A value of α close to 0 emphasizes the attribute information, while a value close to 1 prioritizes the topology. Table 2, shows the α corresponding to the best results, selected on a validation set. We distinguish between WebKB and Planetoid graphs.

For WebKB graphs, we notice that α tends to be higher during the imputation task compared to classification. This suggests an increased emphasis on the graph structure during the imputation optimization and a decreased emphasis during classification. Conversely, the opposite trend is observed for Planetoid graphs. Additionally, on the SBM graph, which has a balanced homophily, α remains relatively stable, hovering around 0.50. Smaller values of α during the imputation of homophilic graphs imply that the structure is considered less crucial than with the heterophilic graphs. However, when it comes to classifying heterophilic graphs, it appears beneficial for the network to downplay the importance of structure, as it can potentially lead to misleading outcomes.

Moreover, we note that α is generally higher for real graphs (and similar for SBM) in the MNAR scenario, indicating a greater emphasis on topology than in the MCAR scenario. We recall that the MNAR mechanisms tend to mask extreme values and attributes of weakly connected nodes. Thus, information loss on attributes caused by the MNAR mechanism is likely being counterbalanced during the network training by leveraging more of the graph's structure.

For homophilic graphs, we observed only a small decrease of AUC when focusing on MAE for the cross-validation (1.22%) against a decrease of 16% on heterophilic ones. We believe that this is a particularly interesting finding as the performances of recently proposed models are mostly evaluated on homophilic graphs using node classification accuracy.

5.3 Imputing Missing Values for Unseen Nodes

The key feature of our approach is the ability to impute missing values on new nodes dynamically added to the graph without having to retrain our imputer from scratch. To evaluate this feature, we propose to build a test set composed of nodes that were fully removed from the graph during the training of `GRIOT`.We report results in Figure 3 and focus on the most heterophilic graphs, as we have shown that these are the most complex ones. For the MAE, we observe that the results of our imputer on unseen nodes are consistent with the ones obtained when all nodes are known from training time, with a small increase in MAE only for extreme cases (80% of missing data). However, this increase in MAE does not coincide with a decrease in AUC score. Indeed, for AUC the results are comparable most of the time, and the imputer is even getting slightly better results on unseen nodes in the majority of the scenarios.

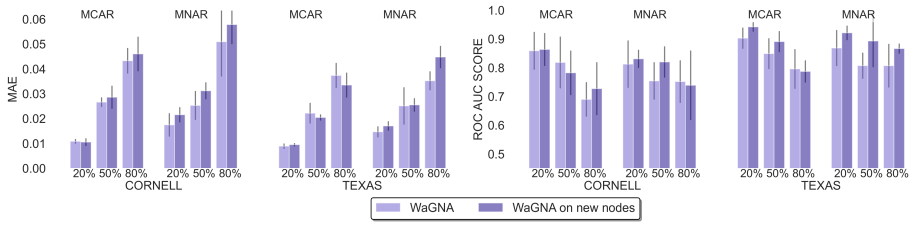

Fig. 3. Performance in terms of (left) MAE and (right) AUC score of GRIOT and GRIOT tested on nodes not present at training time for all settings.

Table 3. Imputation time complexity (in seconds) for all graphs. We report results for the case of 50% missing data.

Model	CITESEER	CORA	PUBMED	CORNELL	TEXAS	WISCONSIN
GRIOT-CPU	846 ± 162	338 ± 6	1096 ± 216	186 ± 10	195 ± 64	244 ± 67
GRIOT-GPU	571 ± 48	236 ± 6	819 ± 189	194 ± 18	148 ± 44	168 ± 67
FP	14 ± 1	4 ± 0	18 ± 1	$.3 \pm .0$	$.3 \pm .1$	$.3 \pm .0$
OT-TAB	489 ± 17	160 ± 7	597 ± 32	79.2 ± 1	78.0 ± 0	43 ± 0
OT-TAB-rr (estimation)	$5.5e6$	$9.0e5$	$8.3e5$	$1.7e5$	$1.6e5$	$1.9e5$

5.4 Time Complexity

Table 3 displays the running times. GRIOT is comparable in processing time to OT-TAB [23], which is less complex as it does not involve imputer training and fills missing values in one step. Additionally, we were not able to compare with OT-TAB-rr, based on Round-Robin, as running time was always exceeding multiple days per run. Unsurprisingly, GRIOT is considerably slower than FP, which does not require any learning process. Finally, FP is faster than GRIOT as it does not require training an imputer (one-shot imputation). Bearing this in mind and taking CITESEER as an example, it means that in a dynamic environment GRIOT becomes more efficient than FP if more than 40 new nodes requiring attribute imputation appear in the graph after training, and more efficient than OT-TAB after just 1 new node.

Summing up our Results. When summarizing the comparison between GRIOT and state-of-the-art methods (Table 4), we observe that GRIOT significantly enhances data reconstruction in 68% of the scenarios and improves the classification task in 37% of the cases. However, it remains on par with the best-performing method, FP if we consider its impact on node classification. Our method is particularly relevant in dynamic environments such as social graphs, where the number of new users is constantly increasing. The use of a trained imputer improves adaptability and guarantees our efficiency in dynamic environments. Furthermore, we hope that our findings provide interesting insights to the community regarding the counterintuitive observations made between the quality of imputed values and subsequent node classification, offering the potential for

Table 4. Each cell shows the number of times GRIOT (a) underperforms, (b) shows no significant changes, or (c) outperforms compared to strong baselines when averaged over all scenarios.

	(a) underperform	(b) similar	(c) outperform
MAE	6%	26%	68%
ROC	8%	56%	37%

valuable advances in the understanding and optimization of imputation strategies for various applications.

6 Conclusion and Perspectives

We introduce GRIOT, a framework employing OT and the MultiW metric for missing attribute imputation in attributed graphs. Key features include support for multiple graph representations, efficient parallelization of the imputation, and a trained imputer for new nodes. Finally, GRIOT is also independent of the task at hand and can therefore be used for tasks other than node classification. Our extensive experiments, spanning synthetic and real-world data with diverse missingness mechanisms, demonstrate the competitiveness of GRIOT in addressing the challenge of missing attributes, particularly in weakly homophilic graphs. Another way to better adapt to heterophilic graphs would be to use an imputer architecture that relies less on the homophilic assumption than GCN. We have begun to investigate the potential use of convolutional graph transformers [8] in place of GCN, but have so far been unable to achieve better results. The future perspective of this work will also consist in deepening our study of the impact of different structural views for tasks other than node classification.

Ethical Statement

Datasets used in this article are publicly accessible, and the code is public, making the results reproducible*. This paper presents work in machine learning, a field that can have potential societal consequences, but none of which we feel must be specifically highlighted here.

7 Acknowledgment

This work has been funded by a public grant from the French National Research Agency (ANR) under the "France 2030" investment plan, which has the reference EUR MANUTECH SLEIGHT - ANR-17-EURE-0026.

References

1. Berg, R.v.d., Kipf, T.N., Welling, M.: Graph convolutional matrix completion. arXiv preprint arXiv:1706.02263 (2017)
2. Brogat-Motte, L., Flamary, R., Brouard, C., Rousu, J., d'Alché-Buc, F.: Learning to predict graphs with Fused Gromov-Wasserstein barycenters. In: ICML. pp. 2321–2335 (2022)
3. Chen, X., Chen, S., Yao, J., Zheng, H., Zhang, Y., Tsang, I.W.: Learning on attribute-missing graphs. IEEE PAMI **44**(2), 740–757 (2022)
4. Chung, F.R.: Spectral graph theory, vol. 92. American Mathematical Soc. (1997)
5. Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S.: Learning to extract symbolic knowledge from the World Wide Web. AAAI/IAAI **3**(3.6), 2 (1998)
6. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. Advances in neural information processing systems **26** (2013)
7. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. Advances in neural information processing systems **29** (2016)
8. Dwivedi, V.P., Bresson, X.: A generalization of transformer networks to graphs. CoRR **abs/2012.09699** (2020), https://arxiv.org/abs/2012.09699
9. Flamary, R., Courty, N., Gramfort, A., Alaya, M.Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N.T., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D.J., Tavenard, R., Tong, A., Vayer, T.: Pot: Python optimal transport. Journal of Machine Learning Research **22**(78), 1–8 (2021)
10. Huisman, M.: Imputation of Missing Network Data: Some Simple Procedures. In: Encyclopedia of Social Network Analysis and Mining, pp. 707–715 (2014)
11. Jiang, B., Zhang, Z.: Incomplete graph representation and learning via partial graph neural networks. arXiv preprint arXiv:2003.10130 (2020)
12. Kantorovich, L.V.: Mathematical methods of organizing and planning production. Management science **6**(4), 366–422 (1960)
13. Kerdoncuff, T., Emonet, R., Perrot, M., Sebban, M.: Optimal tensor transport. In: AAAI Conference on Artificial Intelligence. vol. 36, pp. 7124–7132 (2022)
14. Khanam, K.Z., Srivastava, G., Mago, V.: The homophily principle in social network analysis: A survey. Multimedia Tools and Applications **82**(6), 8811–8854 (2023)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
16. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (2017)
17. Kossinets, G.: Effects of missing data in social networks. Social Networks pp. 247–268 (2006)
18. Laclau, C., Redko, I., Choudhary, M., Largeron, C.: All of the fairness for edge prediction with optimal transport. In: AISTATS. vol. 130, pp. 1774–1782 (2021)
19. Little, R.J., Rubin, D.B.: Statistical analysis with missing data, vol. 793. John Wiley & Sons (2019)
20. McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: Homophily in social networks. Annual review of sociology **27**(1), 415–444 (2001)
21. Mensch, A., Blondel, M., Peyré, G.: Geometric losses for distributional learning. In: International Conference on Machine Learning. pp. 4516–4525 (2019)

22. Monti, F., Bronstein, M., Bresson, X.: Geometric matrix completion with recurrent multi-graph neural networks. NIPS **30** (2017)
23. Muzellec, B., Josse, J., Boyer, C., Cuturi, M.: Missing data imputation using optimal transport. In: ICML. pp. 7130–7140 (2020)
24. Rossi, E., Kenlay, H., Gorinova, M.I., Chamberlain, B.P., Dong, X., Bronstein, M.M.: On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features. In: Learning on Graphs Conference (2022)
25. Schafer, J.L., Graham, J.W.: Missing data: our view of the state of the art. Psychological methods **7**(2), 147 (2002)
26. Stekhoven, D.J., Bühlmann, P.: MissForest—non-parametric missing value imputation for mixed-type data. Bioinformatics **28**(1), 112–118 (2011)
27. Taguchi, H., Liu, X., Murata, T.: Graph convolutional networks for graphs containing missing features. Future Generation Computer Systems **117**, 155–168 (2021)
28. Van Buuren, S., Groothuis-Oudshoorn, K.: mice: Multivariate imputation by chained equations in r. Journal of statistical software (2011)
29. Vayer, T., Chapel, L., Flamary, R., Tavenard, R., Courty, N.: Fused Gromov-Wasserstein distance for structured objects: Theoretical foundations and mathematical properties. Algorithms (2020)
30. Villani, C., et al.: Optimal transport: old and new, vol. 338. Springer Science & Business Media (2009)
31. Yang, Z., Cohen, W., Salakhudinov, R.: Revisiting semi-supervised learning with graph embeddings. In: ICML. pp. 40–48 (2016)
32. Yoon, J., Jordon, J., van der Schaar, M.: GAIN: Missing data imputation using generative adversarial nets. In: Proceedings of ICML. vol. 80, pp. 5689–5698 (2018)
33. Zhang, J., Xiao, X., Huang, L.K., Rong, Y., Bian, Y.: Fine-tuning graph neural networks via graph topology induced optimal transport. In: IJCAI. pp. 3730–3736 (2022)
34. Zheng, X., Liu, Y., Pan, S., Zhang, M., Jin, D., Yu, P.S.: Graph neural networks for graphs with heterophily: A survey. arXiv preprint arXiv:2202.07082 (2022)