GITD: ENHANCING MEDICAL CLASSIFICATION ON TABULAR DATA WITH MISSING VALUES VIA GRAPH MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

With the advancement of machine learning, various techniques have been developed to classify patients for disease diagnosis using medical tabular data. Due to the presence of missing values in the medical tabular data, these techniques commonly impute the missing values before applying classifiers. However, most existing techniques classify patients solely based on each patient's individual features despite the advantages of leveraging patients with similar features that can enhance both imputation and classification. To address this issue, we introduce graph data imputation for tabular data (GITD), a novel approach that constructs feature-attentive k-nearest neighbor (kNN) graphs to enable the use of graph data imputation methods on medical tabular data. The key idea of GITD is constructing a kNN graph among patients by prioritizing important features for classification. Our extensive experimental results demonstrate that GITD successfully bridges graph data imputation methods and medical tabular classification, achieving state-of-the-art performance across various medical tabular datasets.

1 INTRODUCTION

Recent progress in machine learning technology has led to substantial strides in the medical domain (Kononenko, 2001; Giger, 2018; Shehab et al., 2022). Among various types of data in the medical domain, tabular data is one of the most representative forms, consisting of numerical and categorical features for each patient. Many researchers have utilized machine learning frameworks on medical tabular data to classify patients for disease diagnosis (Rahman & Davis, 2013; Liu et al., 2023). The main challenge in handling medical tabular data is that it often contains missing values due to various factors, such as private concerns or incomplete data collection. In this paper, we tackle the classification of patients on medical tabular data with missing values.

To handle medical tabular data containing missing values, imputation techniques that fill in the missing values must be applied prior to classifiers. This is because most classifiers assume that the data is fully observed. Traditionally, simple imputation techniques, such as zero and mean imputation, have been widely used for medical tabular data (Graham et al., 1997; Schafer & Graham, 2002). Recently, deep learning-based imputation techniques (Mattei & Frellsen, 2019; You et al., 2020; Zhong et al., 2023) have demonstrated powerful performance on tabular data, making them an effective approach for medical tabular data. After filling in missing values through the imputation methods, a Multi-Layer Perceptron (MLP) is commonly employed to classify each patient based on the complete data (Sivasankari et al., 2022; Levin et al., 2022).

Medical tabular data typically contains two types of features (Remeseiro & Bolon-Canedo, 2019): (1) class-discriminative features that differentiate among classes and (2) non-discriminative features that have the same distribution regardless of class. For instance, in Alzheimer's disease data (Petersen et al., 2010), the score of a logical memory test can be a key discriminative feature for identifying the disease, while many non-discriminative features, such as the years from the first measurement and the site where data was collected, also exist. Specifically, patients with a particular disease tend to have similar class-discriminative features. Therefore, the classification of a patient can be aided by considering patients who have class-discriminative features similar to those of the patient in question.

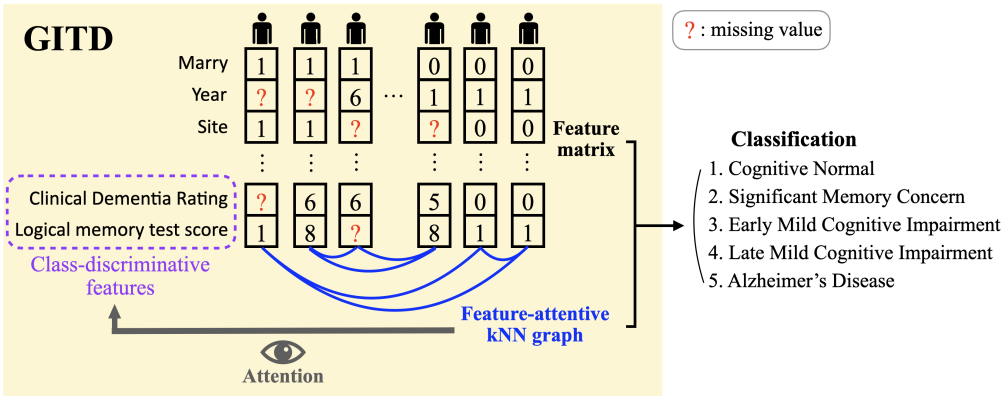


Figure 1: Our graph data imputation for tabular data (GITD) enables the use of graph data imputation methods by constructing a kNN graph primarily based on class-discriminative features. This constructed graph is provided to graph data imputation methods, transferring their outstanding performance to the medical domain and resulting in superior performance compared to existing methods.

Meanwhile, imputation methods developed for graph-structured data (Taguchi et al., 2021; Rossi et al., 2022; Um et al., 2023) have garnered significant attention due to their remarkable effectiveness in handling high rates of missing values. In many real-world graph-structured datasets, there is homophily, which refers to the tendency for nodes to be connected when they belong to the same class or have similar feature values. Based on homophily (McPherson et al., 2001), graph data imputation methods leverage the valuable information in each node's neighbors, leading to outstanding performance in downstream tasks. Although tabular medical data does not have predefined connectivity among patients, connecting patients with similar class-discriminative features can promote homophily, where connected patients are more likely to belong to the same class.

To this end, we propose a novel scheme called graph data imputation for tabular data (GITD), which constructs a kNN graph on medical tabular data with a focus on class-discriminative features. GITD first trains a feature-wise attention network to infer the influence of each feature value on classification. Using the trained network, GITD calculates each feature's importance to classification, considering all patients. GITD then performs kNN graph construction based on the feature importance. Finally, by introducing the constructed kNN graph to graph data imputation models, we can train these models for classification tasks on medical tabular data. Despite its simplicity, GITD significantly improves the classification performance of graph data imputation methods compared to existing kNN graph construction algorithms. Extensive experimental results demonstrate the superiority of GITD using graph data imputation methods in disease diagnosis across various real-world medical tabular datasets, achieving state-of-the-art performance over existing tabular imputation methods.

The main contributions of our work are summarized as:

- To the best of our knowledge, this work is the first attempt to apply graph data imputation methods to tabular data.
- We introduce GITD, which bridges graph data imputation methods and medical tabular data. Based on the nature of medical tabular data, GITD builds a kNN graph that is attentive to class-discriminative features.
- We demonstrate that graph data imputation methods using feature-attentive kNN graphs significantly outperform existing state-of-the-art methods in medical classification and GITD can also provide valuable medical insights.

2 Related Work

2.1 Tabular Data Imputation

Since missing data is a pervasive problem across various domains, handling missing data has long been a prominent area of research in machine learning (Allison, 2009; Lin & Tsai, 2020). For miss-

ing data imputation on tabular data, simple imputation methods such as zero imputation (Schafer & Graham, 2002), mean imputation (Graham et al., 1997), and kNN imputation (Troyanskaya et al., 2001), as well as statistical methods (Van Buuren & Groothuis-Oudshoorn, 2011), have been widely used. With the advancement of deep learning models, deep learning-based approaches have gained popularity due to their effectiveness for accurate imputation. GAIN (Yoon et al., 2018) adopts a Generative Adversarial Nets (GAN) (Goodfellow et al., 2014) framework to generate missing values in tabular datasets. MIWAE (Mattei & Frellsen, 2019) is a framework that enhances Importance-Weighted AutoEncoder (IWAE) (Burda et al., 2015) by introducing a lower bound on the likelihood of observed data to the original objective of IWAE. Recently, graph-based imputation methods, including GRAPE (You et al., 2020) and IGRM (Zhong et al., 2023), have been proposed. These graph-based methods transform a given tabular dataset into a bipartite graph, where nodes consist of sample nodes and feature nodes. By predicting the edge weight between a sample node and a feature node on this bipartite graph, the graph-based methods estimate missing values in the tabular dataset. To perform classification tasks after imputation processes, sample-level classifiers, such as an MLP classifier, are commonly applied to the data completed by various imputation techniques. However, sample-level classifiers cannot leverage the relationships among samples, which can play a crucial role in classification tasks.

2.2 Graph Data Imputation

Several methods tackle the reconstruction of missing values in graph-structured data by minimizing the reconstruction error between the observed values and their reconstructed values. Since node classification is a primary task in graph learning, many approaches have been developed to address node classification with missing values rather than focusing on the accurate reconstruction of missing values. These approaches can be categorized into graph neural network (GNN) architecture-based methods and propagation-based methods. GNN architecture-based methods, including GCNMF (Taguchi et al., 2021) and PaGNN (Jiang & Zhang, 2020), propose new GNN architectures to learn graph-structured data with partially observed feature values. Propagation-based methods, including FP (Rossi et al., 2022) and PCFI (Um et al., 2023), impute missing values through the iterative propagation of observed values on a graph. While preserving the observed values, these methods update missing values by repeatedly aggregating values from neighboring nodes. After imputation, propagation-based methods, employ GNN to perform node classification tasks. In summary, graph data imputation methods commonly incorporate GNNs that are trained on the training samples. While graph data imputation frameworks require pre-defined connectivity among samples, our GITD makes graph data imputation frameworks to tabular data classification by building a new graph primarily based on class-discriminative features.

3 Proposed Method

3.1 Problem Setup

We consider a medical tabular dataset containing missing feature values. We let $\mathbf{X}_{og} \in \mathbb{R}^{N \times F_{og}}$ be the feature matrix of the given medical tabular dataset, where N and F_{og} denote the number of samples (patients) and the number of features, respectively. $\mathbf{M}_{og} \in \{0,1\}^{N \times F_{og}}$ denotes a binary mask where values of 1 indicate the location of missing values. These features consist of numerical and categorical features. To employ imputation techniques, we convert each categorical feature into its dummy variables. This process yields $\mathbf{X} \in \mathbb{R}^{N \times F}$ and $\mathbf{M} \in \{0,1\}^{N \times F}$ from $\mathbf{X}_{og}$ and $\mathbf{M}_{og}$, respectively, where F represents the sum of the number of given numerical features and the number of dummy variables. Let $\mathbf{Y} = [y_1, \ldots, y_N]^\top$ be the labels of samples and $y_i \in \{1, \ldots, C\}$, where y_i denotes the disease-related class label of the i-th sample and C denotes the number of classes. We assume that labels are given for only a subset of the samples (*i.e.*, the training samples). The remaining samples, which are not used for training, are unlabeled (*i.e.*, the validation and test samples). The goal of medical classification is to predict the classes of the test samples based on $\mathbf{X}$, which contains missing values and the partially available labels for the training samples.

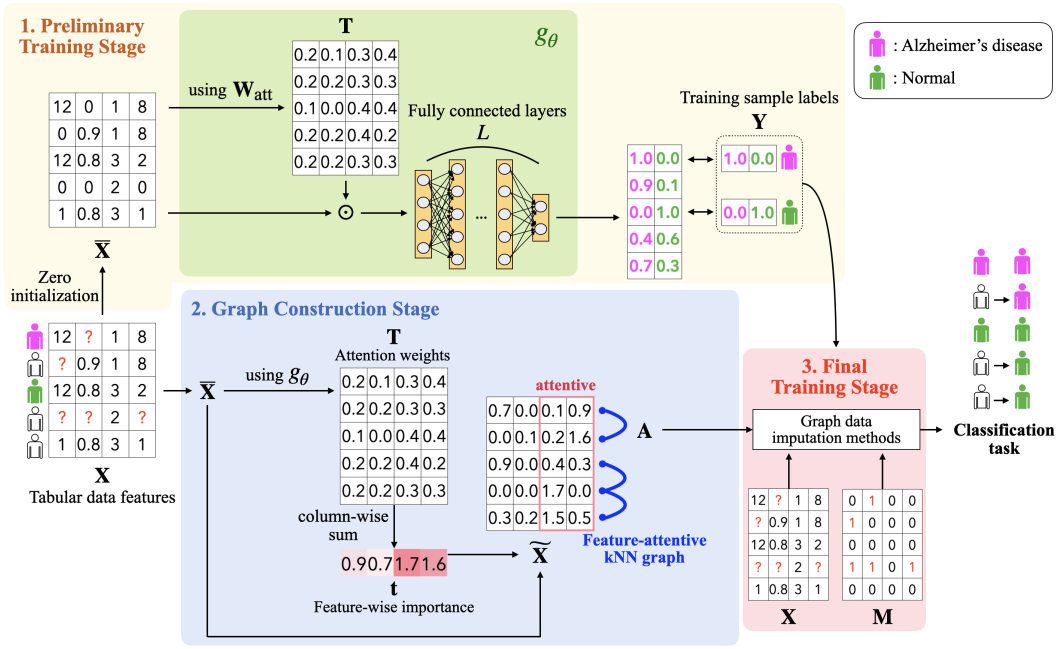


Figure 2: A brief overview of GITD: In the preliminary training stage, g_θ, an MLP classifier with an attention mechanism, is first trained using supervised learning. In the graph construction stage, we utilize the trained g_θ to compute $\mathbf{t}$, which represents feature-wise importance. Using $\mathbf{t}$, this stage constructs a kNN graph that focuses on class-discriminative features. Finally, $\mathbf{A}$, the adjacency matrix of the kNN graph, is provided to the final training stage, enabling graph data imputation methods to be trained and to perform a classification task.

3.2 Overview of GITD

We propose a novel approach called raph data imputation for tabular data (GITD), designed to adapt graph data imputation techniques for medical tabular data containing missing values. While graph data imputation methods require predefined connectivity among data points, medical tabular data typically lacks inherent connectivity. To transfer the powerful performance of graph data imputation to the medical tabular domain, GITD constructs the connectivity among patients. GITD is designed to construct this graph structure mainly based on class-discriminative features to assist graph data imputation methods in performing classification.

Figure 2 provides a brief overview of GITD. The process of GITD consists of three stages: a preliminary training stage, a graph construction stage, and a final training stage. In the preliminary training stage, g_θ, an MLP classifier with an attention mechanism, is trained using supervised learning. After training, all samples are passed through the trained g_θ to obtain the feature-wise attention weights for each sample. These attention weights are then summed across the samples in a feature-wise manner, producing feature-wise attention weights. In the graph construction stage, a kNN graph is built using weighted cosine similarity based on the feature-wise attention weights, making the kNN graph attentive to class-discriminative features. In the final training stage, using this kNN graph, graph data imputation models are trained utilizing GNN-based frameworks to classify samples.

3.3 Preliminary Training Stage

Given $\mathbf{X} \in \mathbb{R}^{N \times F}$, the feature matrix of a medical tabular dataset, we first produce $\overline{\mathbf{X}} \in \mathbb{R}^{N \times F}$ from $\mathbf{X}$ by imputing missing values with zeros. Using $\overline{\mathbf{X}}$, the preliminary training stage then trains g_θ, an MLP classifier with an attention mechanism. Specifically, we compute attention weights

$\mathbf{T} \in \mathbb{R}^{N \times F}$ as follows:

$$\mathbf{T}_{i,j} = \frac{\exp\left(((\overline{\mathbf{X}}_{i,:})\mathbf{W}_{\text{att}})_j\right)}{\sum_{k=1}^{F} \exp\left(((\overline{\mathbf{X}}_{i,:})\mathbf{W}_{\text{att}})_k\right)}, \tag{1}$$

where $\mathbf{W}_{\text{att}} \in \mathbb{R}^{F \times F}$ is a trainable weight matrix and $\overline{\mathbf{X}}_{i,:}$ denotes the i-th row of $\overline{\mathbf{X}}$. Here, $\mathbf{T}_{i,j}$ represents the attention weight for the j-th feature of the i-th sample, and the softmax function ensures that the sum of attention weights across all features for each sample equals 1 (*i.e.*, $\sum_{j=1}^{F} \mathbf{T}_{i,j} = 1$). We then apply the attention weights $\mathbf{T}$ to $\overline{\mathbf{X}}$ as follows:

$$\overline{\mathbf{X}}^{\text{att}} = \overline{\mathbf{X}} \odot \mathbf{T}, \tag{2}$$

where $\odot$ denotes element-wise multiplication.

g_θ, consisting of L layers, then processes the attention-weighted feature matrix $\overline{\mathbf{X}}^{\text{att}}$ through a series of fully connected layers in a sample-wise manner, applying linear transformations followed by non-linear activations such as ReLU. Formally,

$$\mathbf{H}^{(l)} = \sigma(\mathbf{H}^{(l-1)}\mathbf{W}^{(l)} + \mathbf{b}^{(l)}), \quad l = 1, \ldots, L-1 \tag{3}$$

where $\mathbf{H}^{(l)}$ represents the output of the l-th layer, $\mathbf{W}^{(l)} \in \mathbb{R}^{d_{l-1} \times d_l}$ and $\mathbf{b}^{(l)} \in \mathbb{R}^{d_l}$ are the weight matrix and bias vector of the l-th layer, respectively, and $\sigma(\cdot)$ denotes the activation function (*e.g.*, ReLU). Here, $\mathbf{H}^{(0)} = \overline{\mathbf{X}}^{\text{att}}$, the input to the first layer, with $d_0 = F$.

This process continues through all the hidden layers until the final layer, where the output logits $\hat{\mathbf{Y}} \in \mathbb{R}^{N \times C}$ are computed as:

$$\hat{\mathbf{Y}} = \mathbf{H}^{(L-1)}\mathbf{W}^{(L)} + \mathbf{b}^{(L)}, \tag{4}$$

where $\mathbf{W}^{(L)} \in \mathbb{R}^{d_{L-1} \times C}$ and $\mathbf{b}^{(L)} \in \mathbb{R}^{C}$ are the weight matrix and bias vector of the output layer, respectively. Finally, a softmax function is applied to the logits for each sample to produce the predicted class probabilities. g_θ is trained using cross-entropy loss, computed between the one-hot encoded labels from the training sample labels $\mathbf{Y}$ and the predicted probabilities from $\hat{\mathbf{Y}}$.

3.4 Graph Construction Stage

After g_θ is trained through the training stage, $\mathbf{T}$ can provide the importance of each feature for each sample. Thus, we feed $\overline{\mathbf{X}}$ into the trained g_θ and obtain the attention weights $\mathbf{T}$. Since GITD requires feature-wise importance representing the degree to which each feature contributes to classifying the classes, we calculate the feature-wise importance $\mathbf{t} \in \mathbb{R}^F$ by summing $\mathbf{T}$ across the samples as $\mathbf{t}_j = \sum_{i=1}^{N} \mathbf{T}_{i,j}$. This feature-wise importance $\mathbf{t}$ reflects the degree to which each feature is class-discriminative.

To construct a kNN graph using $\mathbf{t}$, we first normalize each sample in $\overline{\mathbf{X}}$ and weight the features according to $\mathbf{t}$ as follows:

$$\widetilde{\mathbf{X}}_{i,j} = (\mathbf{t}_j)^\alpha \cdot \frac{\overline{\mathbf{X}}_{i,j}}{\|\overline{\mathbf{X}}_{i,:}\|_2}, \quad \text{for } i = 1, \ldots, N, \tag{5}$$

where $\widetilde{\mathbf{X}} \in \mathbb{R}^{N \times F}$, $\alpha > 0$ is a hyperparameter that controls the influence of feature importance, and $\|\overline{\mathbf{X}}_{i,:}\|_2$ is the L2 norm of the i-th row of $\overline{\mathbf{X}}$.

Given an arbitrary matrix $\mathbf{B} \in \mathbb{R}^{a \times b}$, we define $\text{kNN}(\cdot) : \mathbb{R}^{a \times b} \to \{0,1\}^{a \times a}$ as a function that generates an adjacency matrix of the row-wise kNN graph (*i.e.*, the kNN graph among rows) based on cosine similarity. We build the kNN graph among samples by

$$\mathbf{A} = \text{kNN}(\widetilde{\mathbf{X}}), \tag{6}$$

where $\mathbf{A} \in \{0,1\}^{N \times N}$ denotes the connections among samples, with values of 1 indicating connected samples. Since $\widetilde{\mathbf{X}}$ is calculated using $\mathbf{t}$, $\mathbf{A}$ can be constructed with a primary focus on class-discriminative features rather than non-discriminative ones.

Table 1: Classification results measured by Micro-F1 score (%). Standard deviation errors are given. OOM denotes an out-of-memory error.

Method	Echocardiogram	Duke Breast Cancer	ABIDE	ADNI QT-PAD	ADNI TADPOLE	Diabetes
Zero	75.33±3.06	74.80±3.91	91.30±0.54	78.22±0.99	77.94±1.24	53.66±0.77
Mean	73.00±4.88	72.76±7.02	68.43±2.13	78.35±1.53	76.89±1.49	53.76±0.33
kNN	77.00±3.71	76.53±0.82	90.45±1.17	80.39±1.30	76.68±1.07	53.92±0.86
GAIN	68.67±4.99	76.31±1.32	89.30±1.81	77.46±1.22	75.46±1.22	53.58±0.59
MIWAE	69.43±6.25	OOM	64.33±0.93	OOM	OOM	OOM
GRAPE	75.00±0.81	OOM	91.61±0.89	OOM	OOM	OOM
IGRM	69.33±8.21	OOM	66.38±1.85	OOM	OOM	OOM
GITD (ours)	**89.00**±2.71	**76.62**±0.67	**91.69**±1.14	**83.75**±0.99	**80.01**±1.22	**54.65**±1.11

3.5 Final Training Stage

Although the given medical dataset does not have any predefined connectivity, GITD can provide $\mathbf{A}$ to graph data imputation methods that require the connectivity among samples as well as a feature matrix $\mathbf{X}$ and a mask $\mathbf{M}$ indicating the location of missing values. Thus, $\mathbf{A}$, the output of the graph construction stage, enables the use of graph data imputation methods on medical tabular data. GNN models in these methods can then be trained to perform classification on the samples, transferring their powerful performance from the graph domain to the medical tabular domain.

4 Experiments

4.1 Datasets

We conduct experiments on six medical tabular datasets, all of which initially contain missing data, as follows: Echocardiogram (Asuncion et al., 2007), Duke Breast Cancer (Saha et al., 2018), ABIDE (Di Martino et al., 2014), ADNI QT-PAD (Petersen et al., 2010), ADNI TADPOLE (Petersen et al., 2010), and Diabetes (Asuncion et al., 2007). The datasets have missing data rates of 2.59%, 11.94%, 52.52%, 22.29%, 27.31%, and 4.03%, respectively. Detailed information on these datasets is provided in Appendix B.1.

4.2 Compared Methods

We compare GITD with seven tabular data imputation methods on medical tabular datasets. These methods are categorized into two groups: (1) conventional methods: zero imputation (Schafer & Graham, 2002), mean imputation (Graham et al., 1997), and kNN imputation (Troyanskaya et al., 2001); and (2) state-of-the-art deep learning-based methods: GAIN (Yoon et al., 2018), MIWAE (Mattei & Frellsen, 2019), GRAPE (You et al., 2020), and IGRM (Zhong et al., 2023). For graph data imputation methods, we employ GCNMF (Taguchi et al., 2021), PaGNN (Jiang & Zhang, 2020), FP (Rossi et al., 2022), and PCFI (Um et al., 2023). As the default setting for GITD, FP is utilized as a graph data imputation method. That is, unless otherwise specified, we use FP as the graph data imputation method for GITD.

4.3 Experimental Setup

To evaluate the performance of imputation methods in medical classification with missing data, we compare classification performance on medical tabular data containing missing values. For a fair comparison, we generate five random splits for training, validation, and test samples with proportions of 0.1, 0.1, and 0.8, respectively. To evaluate classification performance, we measure the average Micro-F1 score across the five splits. For the six tabular data imputation methods except for GRAPE, we employ MLP classifiers on the imputed feature matrices to perform classification. Since GRAPE has an integrated version that includes a classifier, we use that version for classification. Graph data imputation methods are categorized into two approaches: (1) single-stage, including GCNMF and PaGNN, and (2) two-stage, including FP and PCFI. While the single-stage methods perform imputation and classification within a single framework, we utilize GCNs (Kipf & Welling, 2016) as downstream GNNs for the two-stage methods. For GITD, we employ grid search

Table 2: Comparison of feature-attentive kNN graph construction and typical graph construction algorithms in terms of Micro-F1 score for medical classification. FC, SIM_FC, kNN, and ATT_kNN represent an unweighted fully connected graph, a fully connected graph with feature similarity weights, a typical kNN graph, and our feature-attentive kNN graph. GITD models with different graph construction algorithms are evaluated. Improvement (%) denotes the improvement percentage, representing the percentage improvement of ATT_kNN over kNN.

Dataset	Echocardiogram	Duke Breast Cancer	ABIDE	ADNI QT-PAD	ADNI TADPOLE	Diabetes
FC	67.67±1.33	77.08±0.70	49.62±1.80	32.34±0.14	28.96±0.57	OOM
SIM_FC	67.67±1.33	77.08±0.70	49.62±1.80	32.34±0.14	28.96±0.57	OOM
kNN	85.67±4.67	75.38±2.82	90.65±1.51	83.31±1.25	78.90±0.94	53.03±0.85
ATT_kNN (ours)	**89.00±2.71**	**76.62±0.67**	**91.69±1.14**	**83.75±0.99**	**80.01±1.22**	**54.65±1.11**
Improvement	**+3.89%**	**+1.64%**	**+0.70%**	**+0.53%**	**+1.41%**	**+3.05%**

Table 3: Classification performance for varying label rates, measured by Micro-F1 score (%). OOM denotes an out-of-memory error.

Dataset	Echocardiogram			Duke Breast Cancer			ABIDE		
Label rate	5%	10%	20%	5%	10%	20%	5%	10%	20%
Zero	67.50±10.15	75.33±3.06	78.11±2.82	75.57±1.29	74.80±3.91	77.48±0.80	88.67±0.74	91.30±0.54	91.17±0.75
Mean	65.00±4.90	73.00±4.88	73.21±6.13	75.39±2.38	72.76±7.02	76.82±1.40	62.96±4.02	68.43±2.13	71.63±1.62
kNN	71.56±13.67	77.00±3.71	76.98±8.72	75.54±2.02	76.53±0.82	77.42±0.68	88.25±0.75	90.45±1.17	91.04±0.65
GAIN	66.25±9.09	68.67±4.99	75.85±4.37	75.28±1.44	76.31±1.32	77.48±0.97	86.32±1.28	89.30±1.81	91.50±0.48
MIWAE	65.94±7.68	69.43±6.25	71.70±3.38	OOM	OOM	OOM	61.14±1.81	64.33±0.93	66.11±0.65
GRAPE	66.88±3.75	75.00±0.81	54.72±15.42	OOM	OOM	OOM	**91.80±0.41**	91.61±0.89	84.57±18.75
IGRM	68.13±7.10	69.33±8.21	72.08±3.85	OOM	OOM	OOM	62.30±4.17	66.38±1.85	72.07±1.83
GITD	**81.56±4.57**	**89.00±2.71**	**84.91±4.13**	**76.30±1.00**	**76.62±0.67**	**77.70±3.04**	89.73±2.32	**91.69±1.14**	**91.78±0.74**

Dataset	ADNI QT-PAD			ADNI TADPOLE			Diabetes		
Label rate	5%	10%	20%	5%	10%	20%	5%	10%	20%
Zero	78.15±1.27	78.22±0.99	79.74±2.21	72.93±2.28	77.94±1.24	80.42±1.44	50.50±3.32	53.66±0.77	53.77±0.98
Mean	78.53±1.65	78.35±1.53	79.08±2.61	71.94±3.22	76.89±1.49	79.44±2.00	52.37±2.11	53.76±0.33	53.81±0.67
kNN	79.17±1.20	80.39±1.30	80.77±1.53	72.36±1.80	76.68±1.07	79.42±0.97	51.65±2.74	53.92±0.86	54.42±0.28
GAIN	78.76±1.46	77.46±1.22	79.00±1.03	73.07±0.94	75.46±1.22	80.67±1.16	50.60±3.37	53.58±0.59	53.94±0.74
MIWAE	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
GRAPE	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
IGRM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
GITD	**83.87±0.58**	**83.75±0.99**	**85.39±0.90**	**77.75±1.68**	**80.01±1.22**	**81.77±0.92**	**53.85±0.50**	**54.65±1.11**	**54.54±1.84**

to tune α in Eq. (5) and k in the kNN graph construction in Eq. (6). k and α are searched within $\{1, 3, 5, 10\}$ and $\{0.25, 0.5, 0.75, 1\}$, respectively, using the validation sets. We provide further details on experiments in Appendix B.

4.4 Comparison with State-of-the-Art Methods

On medical tabular datasets containing initially missing values, we compare the classification performance of GITD against tabular data imputation methods. Table 1 demonstrates the classification performance comparison among the methods, measured by Micro-F1 score (%). As shown in the table, GITD achieves state-of-the-art performance across all datasets. Moreover, the performance gains of our best method over the previous state-of-the-art methods are significant. For example, on Echocardiogram, ADNI QT-PAD, and ADNI TADPOLE, the gains are 15.58%, 4.18%, and 2.66%, respectively. Furthermore, we observe that deep learning-based tabular imputation methods, except for GAIN, suffer from out-of-memory errors, indicating poor scalability. In contrast, our method does not suffer from out-of-memory errors, demonstrating the memory efficiency of GITD.

4.5 Comparison of Feature-Attentive kNN Graph Construction and Existing Graph Construction Algorithms

To investigate the source of GITD's outstanding performance, we conduct experiments comparing GITD with feature-attentive kNN graph construction to GITD with typical graph construction algorithms, including an unweighted fully connected graph (denoted as FC), a fully connected graph

Table 4: Classification performance measured by Micro-F1 score (%). Standard deviation errors are given. OOM denotes an out-of-memory error.

Method	Echocardiogram	Duke Breast Cancer	ABIDE	ADNI QT-PAD	ADNI TADPOLE	Diabetes
GTID using GCNMF	88.33±2.11	76.89±0.94	81.08±14.91	83.31±1.35	78.58±0.67	53.40±1.61
GTID using PaGNN	88.33±1.83	**77.17±1.69**	90.97±2.01	**83.85±0.61**	80.13±1.18	54.41±1.05
GTID using PCFI	87.00±1.94	76.29±1.13	91.19±1.31	83.52±0.65	**80.68±1.48**	52.63±0.96
GTID using FP (default)	**89.00±2.71**	76.62±0.67	**91.69±1.14**	83.75±0.99	80.01±1.22	**54.65±1.11**

with feature similarity weights (denoted as SIM_FC), and a typical kNN graph. Table 2 presents the comparison results. As shown in the table, feature-attentive kNN graph construction (denoted as ATT_kNN) significantly improves the performance of GITD compared to its use with typical graph construction algorithms. Notably, ATT_kNN consistently outperforms typical graph construction algorithms across all datasets. This indicates that feature-attentive kNN graph construction has greatly contributed to adapting graph data imputation methods to medical tabular data, leading to their remarkable performance. Furthermore, it suggests that the superior performance of these methods arises not from simply using a kNN graph, but specifically from utilizing our feature-attentive kNN graph.

4.6 Effect of Label Rate on Performance

Since the preliminary training stage, which affects the graph construction stage, utilizes the labels of training samples, the performance of GITD may be influenced by the proportion of labeled training samples. Therefore, we compare the average Micro-F1 score of GITD and other methods by varying the label rates. Table 3 presents the comparison results for varying label rates. As shown in the table, in all cases except for the ABIDE dataset at a label rate of 5%, GITD consistently achieves state-of-the-art performance over tabular imputation methods. These results validate the robustness of GITD against varying label rates.

4.7 GITD using other graph data imputation methods

As mentioned in Sec. 4.2, we utilize FP as the default setting for the graph data imputation method in GITD. However, other graph data imputation methods, including GCNMF, PaGNN, and PCFI, can also be employed as the graph data imputation method in GITD. To demonstrate that GITD using graph data imputation methods other than FP is also effective in medical classification on tabular datasets, we conduct comparative experiments using GITD with different graph data imputation methods. Table 4 presents the results of the comparative experiments. As shown in the table, GITD models with different graph data imputation methods exhibit competitive classification performance when compared to each other across datasets. The graph data imputation method that achieves the best performance varies depending on the dataset, with each method performing best on a specific dataset. This implies that the outstanding performance of GITD does not stem from the use of FP as a graph data imputation method, and other imputation methods can be used in its place. We select FP as the default graph data imputation method in GITD because GITD using PCFI shows good performance across datasets. However, GITD using other graph data imputation methods also generally achieves state-of-the-art performance when compared to the results of existing tabular data imputation methods presented in Table 1. Thus, replacing FP with other graph data imputation methods does not significantly affect the superiority of GITD.

4.8 Time Complexity Analysis

Here we discuss the time complexity of GITD. The time complexity of feature-attentive kNN graph construction consisting of the two stages, the preliminary training stage and the graph construction stage, is $O\left(N \cdot (F^2 + \sum_{l=1}^{L} d_{l-1} \cdot d_l) + N^2 \cdot F\right)$. We then determine the duration of feature-attentive kNN graph construction by measuring running times. Table 5 shows a comparison of the running times among all the methods com-

Table 5: Running times (seconds). ATT_kNN denotes feature-attentive kNN graph construction.

Dataset	Echocardiogram		ABIDE	
Method	ATT_kNN	Total	ATT_kNN	Total
Zero	-	4.2	-	4.6
Mean	-	4.3	-	4.8
kNN	-	4.2	-	4.6
GAIN	-	8.5	-	13.2
MIWAE	-	6.4	-	20.4
GRAPE	-	200.1	-	721.6
IGRM	-	1683.3	-	1718.4
GTID using GCNMF	1.3	7.2	1.4	10.9
GTID using PaGNN	1.3	5.7	1.4	6.9
GTID using PCFI	1.3	11.2	1.4	12.7
GTID using FP (default)	1.3	5.8	1.4	7.5

Table 6: Memory usage of GITD for different datasets, measured in gigabytes (GB).

Dataset	Echocardiogram	Duke Breast Cancer	ABIDE	ADNI QT-PAD	ADNI TADPOLE	Diabetes
Graph Construction	0.001	0.506	0.054	0.734	0.628	1.326
Total	1.192	1.338	1.251	1.572	1.495	2.213

Table 7: Performance comparison of GITD with different initialization strategies, measured by Micro-F1 score (%).

Dataset	ABIDE		ADNI TADPOLE	
Initialization	Zero (used)	Mean	Zero (used)	Mean
GITD using GCNMF	**81.08**$\pm$**14.91**	78.47$\pm$14.84	**78.58**$\pm$**0.67**	71.99$\pm$9.05
GITD using PaGNN	**90.97**$\pm$**2.01**	75.21$\pm$10.95	**80.13**$\pm$**1.18**	80.01$\pm$1.50
GITD using PCFI	**91.19**$\pm$**1.31**	81.96$\pm$3.96	**80.68**$\pm$**1.48**	80.13$\pm$1.59
GITD using FP	**91.69**$\pm$**1.14**	75.80$\pm$11.05	**80.01**$\pm$**1.22**	79.55$\pm$1.89

pared in this paper. We select the Echocardiogram and ABIDE datasets since deep learning-based tabular data imputation methods lead to out-of-memory errors on the other datasets. We observe that feature-attentive kNN graph construction occupies a relatively small portion of the running times in GITD. Additionally, we confirm that GITD generally take less time compared to deep learning-based tabular imputation methods. In summary, feature-attentive kNN graph construction is a fast algorithm, avoiding any significant time burden on graph data imputation methods. Furthermore, we confirm that GITD are more efficient on medical tabular data compared to existing state-of-the-art methods, as shown in Table 1.

4.9 Memory Complexity Analysis

In the process of feature-attentive kNN graph construction, the memory is utilized for training the model g_θ and constructing the kNN graph. To mitigate the heavy memory usage during kNN graph construction, we leverage a batch-wise kNN graph construction strategy. When constructing kNN graphs among samples, we divide batches with batchsize B, and calculate k-nearest neighbors for each batch. This strategy reduces the memory requirement because it avoids the need to store distances between all samples in the entire dataset at once. Specifically, in terms of memory complexity, batch-wise kNN graph construction changes the typical $O(N^2 \cdot F)$ complexity to $O(B \cdot N \cdot F)$. Therefore, the memory complexity of the feature-attentive kNN graph construction process is $O(\theta) + O(B \cdot N \cdot F) + O(N^2)$, where $O(N^2)$ is required for $\mathbf{A}$. We further measure the memory usage in the feature-attentive kNN graph construction process for each dataset. Table 6 shows the results of the measurement. As shown in the table, feature-attentive kNN graph construction requires only a small amount of memory. Furthermore, we can confirm that the entire process of GITD, including training GNN models in the final training stage, operates with the reasonable memory usage.

4.10 Why Is Zero Initialization Used for $\overline{\mathbf{X}}$?

In the graph construction stage of GITD, we use $\overline{\mathbf{X}}$, obtained from $\mathbf{X}$ by imputing missing values with zeros, *i.e.*, we use zero initialization. To justify this initialization strategy, we conduct comparative experiments using a different initialization strategy. We select mean imputation as the comparison strategy, which is a commonly used strategy for initializing missing values. Mean imputation fills in missing values with the mean of observed values. Table 7 demonstrates the results on the ABIDE and ADNI TADPOLE datasets. As shown in the table, GITD using zero initialization consistently outperforms that using mean initialization across the graph data imputation methods on both datasets. These performance gains with zero initialization are attributed to the inherent characteristics of medical tabular data, which often contains many zero values. For instance, among the observed values in the ABIDE and ADNI TADPOLE datasets, 68.10% and 94.78%, respectively, are zeros. This prevalence of zero values makes zero initialization effective in the graph construction stage of GITD on medical tabular datasets.

4.11 Does Feature-wise Importance $\mathbf{t}$ Really Capture Class-Discriminative Features?

To confirm that the feature-wise importance $\mathbf{t}$ of GITD effectively captures class-discriminative features, we conduct an in-depth analysis of $\mathbf{t}$. We extract the two features with the highest values in $\mathbf{t}$ on the ADNI TADPOLE dataset. The features identified are CDRSB_bl and LDELTOTAL_BL, which represent the total score of Clinical Dementia Rating (CDR) and the Logical Memory II Delayed Recall test, respectively, the latter being part of the Wechsler Memory Scale. To verify that these features are class-discriminative features, we calculate the mean and standard deviation of each feature across classes. Each sample in the ADNI TADPOLE dataset belongs to one of five classes related to cognitive impairment: Cognitively Normal (CN), Significant Memory Concern (SMC), Early Mild Cognitive Impairment (EMCI), Late Mild Cognitive Impairment (LMCI), and Alzheimer's Disease (AD). These classes are ordered according to the increasing severity of cognitive impairment, with AD being the most severe.

Table 8: Mean and standard Deviation of the two features with the highest values in $\mathbf{t}$ across different classes. "Std." denotes standard deviation.

Feature	CDRSB_bl		LDELTOTAL_BL	
Class	Mean	Std.	Mean	Std.
CN	0.003	0.012	0.578	0.146
SMC	0.005	0.015	0.565	0.145
EMCI	0.127	0.076	0.391	0.081
LMCI	0.164	0.091	0.169	0.115
AD	0.443	0.165	0.060	0.082

Table 8 shows the distribution of the two features with the highest values in $\mathbf{t}$. As shown in the table, as the severity of cognitive impairment increases, CDRSB_bl increases while LDELTOTAL_BL decreases. This indicates that the values of CDRSB_bl and LDELTOTAL_BL can significantly aid in distinguishing between classes. Furthermore, $\mathbf{t}$ can provide medical insights into which features are critical for disease diagnosis. Conversely, we examine the two features with the smallest values in $\mathbf{t}$. The features found are Years_bl and SITE, which represent the years from the first measurement and an indicator that denotes the specific clinical site where each participant was enrolled, respectively. Table 9 shows the distribution of these two features. We observe that it is difficult to identify trends related to the severity of cognitive impairment in these two features. In summary, the feature-attentive kNN graph constructed in GITD effectively captures class-discriminative features with $\mathbf{t}$ and makes the generated kNN graph attentive to class-discriminative features.

Table 9: Mean and standard Deviation of the two features with the lowest values in $\mathbf{t}$ across different classes. "Std." denotes standard deviation.

Feature	Years_bl		SITE	
Class	Mean	Std.	Mean	Std.
CN	0.122	0.215	0.101	0.186
SMC	0.072	0.137	0.067	0.055
EMCI	0.184	0.236	0.079	0.118
LMCI	0.107	0.189	0.117	0.211
AD	0.090	0.141	0.089	0.124

Analyses of hyperparameter sensitivity and experimental details are provided in Appendix A and Appendix B, respectively.

5 Conclusion

In this paper, we propose GITD, an innovative approach for medical classification that introduces graph data imputation to medical tabular data, which often contain missing values. Specifically, GITD performs the preliminary training process and computes feature-wise importance, making the kNN graph attentive to class-discriminative features. While graph data imputation methods have not been considered in the tabular domain due to their need for predefined connectivity, GITD provides kNN graphs tailored to these imputation methods. By using these feature-attentive kNN graphs, graph data imputation methods transfer their outstanding performance in the graph domain to the medical domain, resulting in remarkable performance gains over existing tabular imputation methods. We further confirm that the feature-attentive kNN graphs can offer important medical insights. Our work demonstrates the potential for graph data imputation methods to be extended to non-graph-structured data. Furthermore, we believe that our work will contribute to machine learning-based disease diagnosis by significantly improving classification performance.

REFERENCES

Paul D Allison. Missing data. *The SAGE handbook of quantitative methods in psychology*, pp. 72–89, 2009. 2

Arthur Asuncion, David Newman, et al. Uci machine learning repository, 2007. 6, 14

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015. 3

Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 19(6):659–667, 2014. 6, 14

Maryellen L Giger. Machine learning in medical imaging. *Journal of the American College of Radiology*, 15(3):512–520, 2018. 1

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 3

John W Graham, Scott M Hofer, Stewart I Donaldson, David P MacKinnon, and Joseph L Schafer. Analysis with missing data in prevention research. 1997. 1, 3, 6

Bo Jiang and Ziyan Zhang. Incomplete graph representation and learning via partial graph neural networks. *arXiv preprint arXiv:2003.10130*, 2020. 3, 6

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 15

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 6

Igor Kononenko. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine*, 23(1):89–109, 2001. 1

Roman Levin, Valeriia Cherepanova, Avi Schwarzschild, Arpit Bansal, C Bayan Bruss, Tom Goldstein, Andrew Gordon Wilson, and Micah Goldblum. Transfer learning with deep tabular models. *arXiv preprint arXiv:2206.15306*, 2022. 1

Wei-Chao Lin and Chih-Fong Tsai. Missing value imputation: a review and analysis of the literature (2006–2017). *Artificial Intelligence Review*, 53:1487–1509, 2020. 2

Mingxuan Liu, Siqi Li, Han Yuan, Marcus Eng Hock Ong, Yilin Ning, Feng Xie, Seyed Ehsan Saffari, Yuqing Shang, Victor Volovici, Bibhas Chakraborty, et al. Handling missing values in healthcare data: A systematic review of deep learning-based imputation techniques. *Artificial intelligence in medicine*, 142:102587, 2023. 1

Pierre-Alexandre Mattei and Jes Frellsen. Miwae: Deep generative modelling and imputation of incomplete data sets. In *International conference on machine learning*, pp. 4413–4423. PMLR, 2019. 1, 3, 6

Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444, 2001. 2

Ronald Carl Petersen, Paul S Aisen, Laurel A Beckett, Michael C Donohue, Anthony Collins Gamst, Danielle J Harvey, CR Jack Jr, William J Jagust, Leslie M Shaw, Arthur W Toga, et al. Alzheimer's disease neuroimaging initiative (adni) clinical characterization. *Neurology*, 74(3): 201–209, 2010. 1, 6, 14

M Mostafizur Rahman and Darryl N Davis. Machine learning-based missing value imputation method for clinical datasets. In *IAENG Transactions on Engineering Technologies: Special Volume of the World Congress on Engineering 2012*, pp. 245–257. Springer, 2013. 1

Beatriz Remeseiro and Veronica Bolon-Canedo. A review of feature selection methods in medical applications. *Computers in biology and medicine*, 112:103375, 2019. 1

Emanuele Rossi, Henry Kenlay, Maria I Gorinova, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features. In *Learning on graphs conference*, pp. 11–1. PMLR, 2022. 2, 3, 6, 15

Ashirbani Saha, Michael R Harowicz, Lars J Grimm, Connie E Kim, Sujata V Ghate, Ruth Walsh, and Maciej A Mazurowski. A machine learning approach to radiogenomics of breast cancer: a study of 922 subjects and 529 dce-mri features. *British journal of cancer*, 119(4):508–516, 2018. 6, 14

Joseph L Schafer and John W Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002. 1, 3, 6

Mohammad Shehab, Laith Abualigah, Qusai Shambour, Muhannad A Abu-Hashem, Mohd Khaled Yousef Shambour, Ahmed Izzat Alsalibi, and Amir H Gandomi. Machine learning in medical applications: A review of state-of-the-art methods. *Computers in Biology and Medicine*, 145:105458, 2022. 1

SS Sivasankari, J Surendiran, N Yuvaraj, M Ramkumar, CN Ravi, and RG Vidhya. Classification of diabetes using multilayer perceptron. In *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, pp. 1–5. IEEE, 2022. 1

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 15

Hibiki Taguchi, Xin Liu, and Tsuyoshi Murata. Graph convolutional networks for graphs containing missing features. *Future Generation Computer Systems*, 117:155–168, 2021. 2, 3, 6

Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001. 3, 6

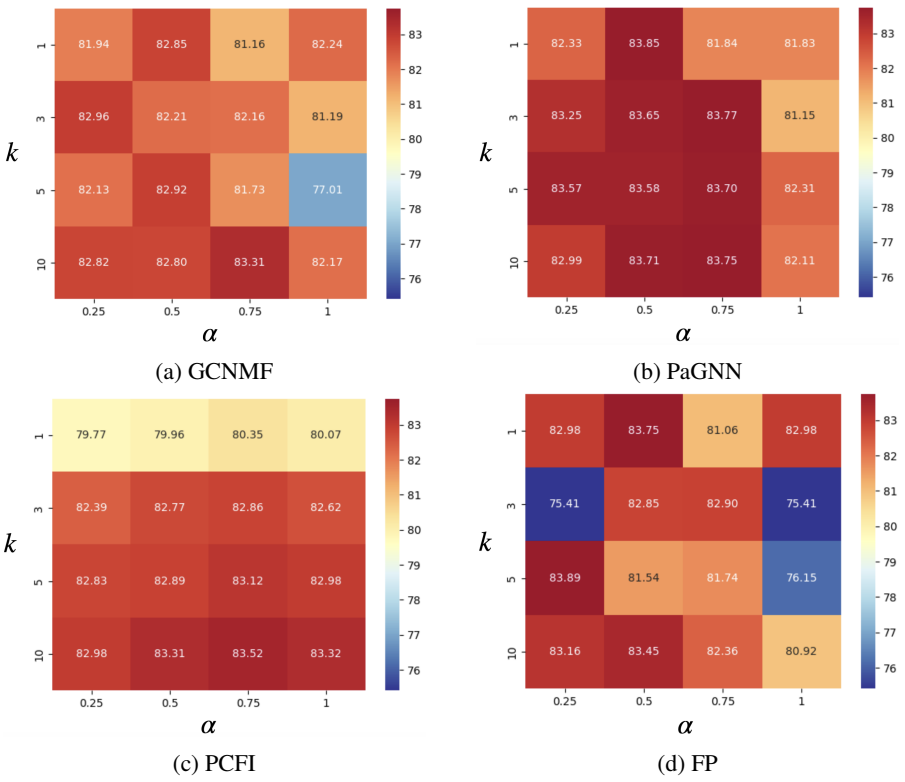
Daeho Um, Jiwoong Park, Seulki Park, and Jin young Choi. Confidence-based feature imputation for graphs with partially known features. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=YPKBIILy-Kt. 2, 3, 6

Stef Van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45:1–67, 2011. 3

Jinsung Yoon, James Jordon, and Mihaela Schaar. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, pp. 5689–5698. PMLR, 2018. 3, 6

Jiaxuan You, Xiaobai Ma, Yi Ding, Mykel J Kochenderfer, and Jure Leskovec. Handling missing data with graph representation learning. *Advances in Neural Information Processing Systems*, 33: 19075–19087, 2020. 1, 3, 6

Jiajun Zhong, Ning Gui, and Weiwei Ye. Data imputation with iterative graph reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11399–11407, 2023. 1, 3, 6



(a) GCNMF (b) PaGNN

(c) PCFI (d) FP

Figure 3: Classification performance of GITD for different k and α on the ADNI QT-PAD dataset, measured by Micro-F1 score (%).

Table 10: Dataset Statistics.

Dataset	N	F	F_{num}	F_{cat}	C	r_m
Echocardiogram	74	12	3	9	2	2.59%
Duke Breast Cancer	907	93	34	59	2	11.94%
ABIDE	1112	104	85	19	2	52.52%
ADNI QT-PAD	1737	96	76	20	5	22.29%
ADNI TADPOLE	2132	110	89	21	5	27.31%
Diabetes	10177	47	11	36	3	4.03%

A Hyperparameter Sensitivity

To investigate the impact of the two hyperparameters of GITD, α and k, we measure the classification performance of GITD using graph data imputation methods by varying α and k on the ADNI QT-PAD dataeset. According to our search ranges for k and α, we vary k and α within $\{1, 3, 5, 10\}$ and $\{0.25, 0.5, 0.75, 1.0\}$, respectively. Figure 3a, Figure 3b, Figure 3c, and Figure 3d demonstrate the classification performance of GITD using graph data imputation methods for different k and α, measured by Micro-F1 score (%). As shown in the figures, the methods generally demonstrate robustness against variations in k and α. Considering the previous state-of-the-art performance of kNN is 80.39%, GITD using graph data imputation methods achieve the state-of-the-art performance with most combinations of (k, α) within the respective search ranges. For example, GITD using PaGNN consistently outperforms the previous state-of-the-art performance, regardless the values of k and α.

B EXPERIMENTAL DETAILS

B.1 DATASET DETAILS

We conduct experiments on six benchmark datasets, including Echocardiogram, Duke Breast Cancer, ABIDE, ADNI QT-PAD and ADNI TADPOLE, and Diabetes. Table 10 presents the statistics of the datasets used in this paper. N and F denote the number of samples and features, respectively. F_{num} and F_{cat} represent the number of numerical features and categorical features, respectively. We transform numerical features by scaling them to a fixed range between 0 and 1. We utilize one-hot encoding for categorical features. While C represents the number of classes, r_m denotes the missing rate of features in a given dataset.

B.1.1 ECHOCARDIOGRAM

The Echocardiogram dataset is a medical tabular dataset related to heart attacks, which can be downloaded from the UCI Machine Learning Repository (Asuncion et al., 2007). The 'alive-at-1' feature, a binary variable, is used as the class label. In this label, 0 indicates that the patient either died within one year or was followed for less than one year, while 1 indicates that the patient was alive at one year. The goal of the classification problem using the Echocardiogram dataset is to predict whether patients will survive for at least one year after a heart attack.

B.1.2 DUKE BREAST CANCER

The Duke Breast dataset is a medical tabular dataset related to breast cancer, available for download from The Cancer Imaging Archive (TCIA) (Saha et al., 2018). The 'Tumor_Grade' feature, which can be one of $\{1, 2, 3\}$, is used as the class label.

B.1.3 ABIDE

The Autism Brain Imaging Data Exchange (ABIDE) dataset is a medical tabular dataset related to autism spectrum disorder, available for download from the ABIDE webpage (Di Martino et al., 2014). The 'DX_GROUP' feature, where 1 and 2 represent autism and control, respectively, is used as the class label.

B.1.4 ADNI QT-PAD AND ADNI TADPOLE

The Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset is a medical tabular dataset used to study the progression of Alzheimer's disease (AD), which can be downloaded from the ADNI webpage (Petersen et al., 2010). We use the 'DB_bl' feature as the class label, which can be one of five cognitive impairment levels: Cognitively Normal (CN), Significant Memory Concern (SMC), Early Mild Cognitive Impairment (EMCI), Late Mild Cognitive Impairment (LMCI), and Alzheimer's Disease (AD). These classes are ordered by increasing severity of cognitive impairment, with AD being the most severe. 'ADNI QT-PAD' and 'ADNI TADPOLE' are located in the tadpole_challenge folder and ADNI_QT-PAD, respectively.

B.1.5 DIABETES

The Diabetes dataset represents the records of hospitalized patients diagnosed with diabetes, which can be downloaded from the UCI Machine Learning Repository (Asuncion et al., 2007). The readmitted feature is the class label, which can be one of the following: 1) '<30': if the patient was readmitted in less than 30 days; 2) '>30': if the patient was readmitted in more than 30 days; 3) 'No': if there was no record of readmission. The goal is to determine whether the patient will be readmitted within 30 days of discharge.

B.2 IMPLEMENTATION DETAILS

We conduct all experiments on a single NVIDIA GeForce RTX 2080 Ti GPU with 11GB of memory and an Intel Core i5-10500 CPU at 3.10GHz. Across all baselines, we adhere to the hyperparameter tuning strategies and settings described in their respective papers. For training graph data imputation

Table 11: Hyperparameter settings of GITD for each graph data imputation method across different datasets.

Method	Hyperparameter	Echocardiogram	Duke Breast Cancer	ABIDE	ADNI QT-PAD	ADNI TADPOLE	Diabetes
GCNMF	k	10	5	10	10	5	5
	α	1	0.25	1	0.75	0.5	1
PaGNN	k	10	10	1	1	10	1
	α	1	1	0.5	0.5	0.25	1
PCFI	k	10	10	1	10	10	1
	α	1	0.5	0.5	0.75	0.75	1
FP	k	10	3	1	1	5	5
	α	0.5	0.25	0.5	0.5	0.25	1

methods used in GITD, we follow (Rossi et al., 2022). We utilize the Adam optimizer (Kingma & Ba, 2014) and set the maximum number of epochs to 10,000. We employ an early stopping strategy based on validation accuracy, with a patience of 200 epochs. Dropout (Srivastava et al., 2014) is applied with a drop probability p, where p is searched within $0, 0.5$. We consistently set the number of GNN layers and the hidden dimension of graph data imputation methods to 2 and 64, respectively. Table 11 shows the hyperparamter settings of GITD for graph data imputation methods across different datasets.