Gene-Gene Relationship Modeling Based on Genetic Evidence for Single-Cell RNA-Seq Data Imputation

Daeho Um*
Samsung Advanced Institute of Technology (SAIT)
daeho.um@samsung.com

Ji Won Yoon
Chung-Ang University
jiwonyoon@cau.ac.kr

Seong Jin Ahn
Korea Advanced Institute of Science and Technology (KAIST)
sja1015@kaist.ac.kr

Yunha Yeo
Korea University
serinahyeo@korea.ac.kr

Abstract

Single-cell RNA sequencing (scRNA-seq) technologies enable the exploration of cellular heterogeneity and facilitate the construction of cell atlases. However, scRNA-seq data often contain a large portion of missing values (false zeros) or noisy values, hindering downstream analyses. To recover these false zeros, propagation-based imputation methods have been proposed using k-NN graphs. However they model only associating relationships among genes within a cell, while, according to well-known genetic evidence, there are both associating and dissociating relationships among genes. To apply this genetic evidence to gene-gene relationship modeling, this paper proposes a novel imputation method that newly employs dissociating relationships in addition to associating relationships. Our method constructs a k-NN graph to additionally model dissociating relationships via the negation of a given cell-gene matrix. Moreover, our method standardizes the value distribution (mean and variance) of each gene to have standard distributions regardless of the gene. Through extensive experiments, we demonstrate that the proposed method achieves exceptional performance gains over state-of-the-art methods in both cell clustering and gene expression recovery across six scRNA-seq datasets, validating the significance of using complete gene-gene relationships in accordance with genetic evidence.

1 Introduction

Single-cell RNA sequencing (scRNA-seq) has become one of the most widely used technologies in biomedical research due to its ability to measure genome-wide gene expression at the single-cell level [1–3]. ScRNA-seq enables us to discover novel cell types [4], analyze cellular trajectories [5], and improve understanding human disease [6, 7]. However, scRNA-seq analysis encounters significant challenges due to the high rate of zero values in scRNA-seq data represented by a cell-gene matrix. Specifically, owing to the low RNA capture rate, scRNA-seq data often contain zero values. These zero values represent unobserved gene expression resulting from both technical omissions (referred to as *dropouts* [8]) and true biological absence. Moreover, even non-zero values in scRNA-seq data suffer from various sources of noise, such as cell cycle effects and batch effects [9, 10].

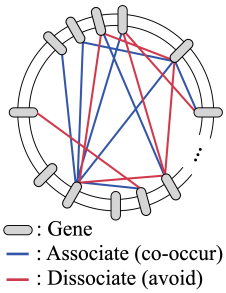

Figure 1: Within a cell, there are two types of relationships among genes.

*Corresponding author

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

To deal with the missing or noisy gene expression in scRNA-seq data, diverse imputation methods have been proposed, which can be categorized into non-graph-based, graph neural network (GNN)-based, and propagation-based methods. Among these methods, propagation-based methods [11, 12] have been favored due to their outstanding performance. The propagation-based methods construct a k-nearest neighbor (k-NN) graph on scRNA-seq data represented as a cell-gene matrix, and fill in missing values by propagating nonzero values on the k-NN graph. Despite their effectiveness, they overlook well-known genetic evidence [13, 14], which means that there are two types of relationships between genes: associating relationship and dissociating one. As shown in Figure 1, associating relationships represent genes that co-occur, whereas dissociating relationships represent genes that avoid co-occurrence.

However, the existing methods cannot model the dissociating gene-gene relationship by constructing a simple k-NN graph to connect only the associating genes with similar occurrence patterns. Consequently, these methods fail to connect dissociating genes. Within a cell, when considering the value to be imputed for gene Q, the value for its associating gene can assist in inferring the value for gene Q. However, its dissociating gene can also provide crucial information: if its dissociating gene has a high value, the value for gene Q may be low, as they tend to avoid each other. Additionally, the value distribution of a gene often differs significantly from that of other genes [15]. Therefore, the sum of propagated values from other genes may lead to the mixing of values at various scales, which is not suitable for data recovery.

To resolve the aforementioned problems, we propose a novel propagation-based imputation scheme called Single-Cell Complete Relationship (scCR) for scRNA-seq data, which models both associating and dissociating gene-gene relationships. scCR concatenates a given cell-gene matrix and its negation, then standardizes the value distribution (mean and variance) in each column (*i.e.*, gene) of the concatenated matrix. Subsequently, we construct a k-NN graph on this concatenated and standardized matrix to connect both associating and dissociating genes within a cell. Through a propagation process on this k-NN graph, scCR effectively denoises scRNA-seq data by capturing complete gene-gene relationships. Extensive experimental results demonstrate that scCR significantly outperforms state-of-the-art methods in both gene expression recovery and cell clustering. Through experiment, we further confirm that scCR can model dissociating gene-gene relationships inherent in scRNA-seq data.

The main contributions of our work are summarized as follows: (1) We newly propose an effective imputation method for scRNA-seq data, which is based on the genetic evidence. Our method can model complete gene-gene relationships, including both associating and dissociating relationships; (2) We employ a standardization step before propagation among genes for additional performance improvement in downstream tasks on scRNA-seq data; (3) By modeling dissociating gene-gene relationships and utilizing the standardization step, our scCR significantly improves performance in various downstream tasks, outperforming the state-of-the-art methods by a large margin.

2 Related Work

Handling noise in scRNA-seq data. Approaches for handling noise in scRNA-seq data can be categorized into non-graph-based, GNN-based, and propagation-based methods. As pioneering efforts to impute zero values, non-graph-based methods predominantly employ either statistical techniques [16, 17] or autoencoder frameworks [17, 18]. Building on this foundation, graph-based approaches, including GNN-based and propagation-based methods, have received significant attention due to their ability to model relationships among cells and genes through graph structures. scGNN [19] leverages cell-cell relationships by constructing a cell-cell similarity matrix within a graph autoencoder framework. scGCL [20] is a graph autoencoder framework that exploits contrastive learning to capture cell-cell relationships. scTAG [21] is a clustering method that employs a graph autoencoder framework using a cell-cell k-NN graph, which jointly optimizes clustering loss and reconstruction loss.

Propagation-based imputation in scRNA-seq data. Propagation-based imputation methods have shown their superiority in scRNA-seq data imputation. They promote greater similarity in gene expression among cells that are already similar through iterative propagation steps. While MAGIC [22] utilizes a diffusion mechanism to denoise scRNA-seq data, updating values through the diffusion of both zero values and observed nonzero values may be significantly affected by false zero values (*i.e.*, dropouts). To address this issue, FP can be a good solution because FP preserves observed values during diffusion while updates unobserved values through diffusion. Propagation-based imputation

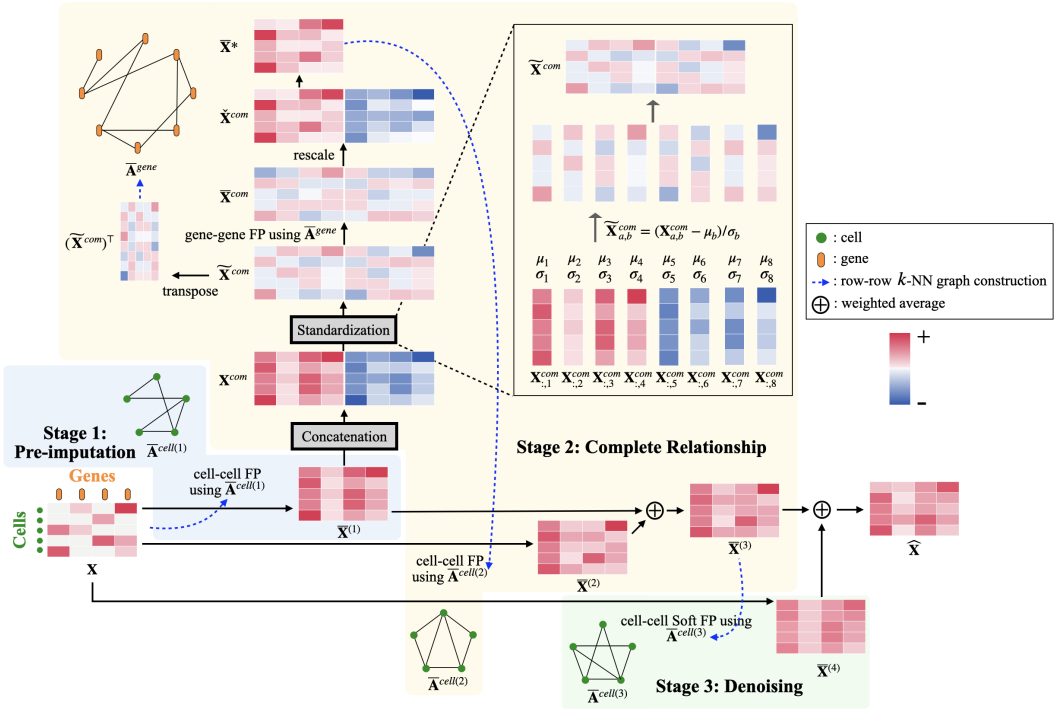

Figure 2: A brief overview of Single-Cell Complete Relationship (scCR).

methods have shown their superiority in scRNA-seq data imputation. scFP [11] adopts FP developed for graph-structured data to resolve imputation for scRNA-seq data. scFP constructs a cell-cell k-NN graph and applies FP for the imputation of zero values. Very recently, [12] proposes scBFP to utilize gene-gene relationships as well as cell-cell relationships. scBFP consists of two stages, and in each stage, it applies FP using a gene-gene k-NN graph and a cell-cell k-NN graph, respectively. Although scBFP is designed to leverage gene-gene relationships, the simple addition of FP using a gene-gene k-NN graph cannot effectively exploit gene-gene relationships due to the following two reasons: **(1)** a gene-gene k-NN graph can connect only associating genes which have co-occurrence relationships while overlooking the presence of dissociating gene-gene relationships; **(2)** Since the distributions for each gene significantly varies, propagation without additional processing will degrade recovery performance.

3 Preliminaries

Notation. A graph can be represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, \ldots, v_N\}$ is the set of N nodes and $\mathcal{E}$ is the set of edges. The connectivity of $\mathcal{G}$ can be represented by the adjacency matrix $\mathbf{A} \in \{0,1\}^{N \times N}$ with $\mathbf{A}_{i,j} = 1$ iff $(v_i, v_j) \in \mathcal{E}$ and $\mathbf{A}_{i,j} = 0$ otherwise. Given an arbitrary matrix $\mathbf{B} \in \mathbb{R}^{a \times b}$, we let $k\text{NN}(\cdot) : \mathbb{R}^{a \times b} \to \{\mathbb{R}\}^{a \times a}$ be a function that generates a normalized adjacency matrix $\overline{\mathbf{A}}$ of the row-row k-NN graph based on cosine similarity. Here, the normalized adjacency matrix $\overline{\mathbf{A}}$ is obtained by $\overline{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ where $\mathbf{A}$ is the adjacency matrix of the k-NN graph and $\mathbf{D}$ is a degree matrix with diagonal entries $\mathbf{D}_{i,i} = \sum_j \mathbf{A}_{i,j}$. Consequently, while $k\text{NN}(\mathbf{X})$ yields $\overline{\mathbf{A}}^{cell} \in \{0,1\}^{C \times C}$ of the cell-cell k-NN graph from $\mathbf{X}$, $k\text{NN}(\mathbf{X}^\top)$ produces $\overline{\mathbf{A}}^{gene} \in \{0,1\}^{G \times G}$ of the gene-gene k-NN graph from $\mathbf{X}$. We let $\mathbf{B}_{i,:}$ and $\mathbf{B}_{:,j}$ denote the i-th row vector of $\mathbf{B}$ and the j-th column vector of $\mathbf{B}$, respectively.

Feature Propagation. Feature Propagation (FP) is proposed to impute missing features in graph-structured data. The core idea of FP is to impute missing values by diffusing observed values while preserving these observed values. Assume that a given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ has a feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$ with missing values, where rows and columns correspond to nodes and F feature channels, respectively. We use $\overline{\mathbf{A}} \in \mathbb{R}^{N \times N}$ to denote a normalized adjacency matrix of the graph. To preserve known features during the diffusion process, we mark the positions of the features to be

preserved with 1 in the mask matrix $\mathbf{M} \in \{0,1\}^{N \times F}$; here, values of 1 in $\mathbf{M}$ indicate the location of observed features. We express FP as a function by $\overline{\mathbf{X}} = \text{FP}(\mathbf{X}, \overline{\mathbf{A}}, \mathbf{M})$ where $\overline{\mathbf{X}} \in \mathbb{R}^{N \times F}$ is an output matrix. A detailed explanation of FP is provided in Appendix A. In summary, FP fills in missing values in $\mathbf{X}$ through diffusion using $\mathbf{A}$ while preserving features corresponding to values of 1 in $\mathbf{M}$. It is noteworthy that $\text{FP}(\mathbf{X}, \mathbf{A}, \mathbf{M})$ performs propagation among the rows of $\mathbf{X}$. In scRNA-seq data imputation, FP-based imputation methods treat zero values as missing values to be imputed via features diffused from non-zero values.

4 Proposed Method

4.1 Overview of scCR

In this paper, we design a novel imputation framework for scRNA-seq data, namely scCR, which utilizes complete gene-gene relationships. Unlike existing work, scCR exploits both associating and dissociating relationships, which contain valuable biological information. Given highly noisy scRNA-seq data, especially having a high number of false zeros, the goal of scCR is to recover scRNA-seq data by imputing zero values. As shown in Figure 2, our proposed framework consists of three stages: pre-imputation, complete relation, and denoising. Throughout these three stages, we enhance a gene expression matrix by gradually integrating complete gene-gene and cell-cell relationships.

4.2 Pre-Imputation Stage

We consider a cell-gene matrix $\mathbf{X} \in \mathbb{R}^{C \times G}$, where C and G represent the number of cells and genes, respectively. We let $\mathbf{A}^{cell} \in \{0,1\}^{C \times C}$ denote an adjacency matrix of a cell-cell graph. Similarly, we let $\mathbf{A}^{gene} \in \{0,1\}^{G \times G}$ denote an adjacency matrix of a gene-gene graph. Building a k-NN graph directly on $\mathbf{X}$ can lead to performance degradation in downstream tasks due to the noisy nature of $\mathbf{X}$. Therefore, scCR begins with the pre-imputation stage, which creates a pre-imputed matrix to be used for k-NN graph construction in the complete relationship stage. In this stage, scCR imputes zero values through intercellular (*i.e.*, cell-cell) propagation.

Cell-cell FP. We first construct a cell-cell k-NN graph by $\overline{\mathbf{A}}^{cell(1)} = k\text{NN}(\mathbf{X})$. scCR then employs FP to impute zero values by the diffusion of nonzero values among cells. We let $\mathbf{M}^{\mathbf{X}} \in \{0,1\}^{C \times G}$ be a mask matrix with $\mathbf{M}^{\mathbf{X}}_{i,j} = 1$ iff $\mathbf{X}_{i,j} \neq 1$ and $\mathbf{M}^{\mathbf{X}}_{i,j} = 0$ otherwise, which indicates the positions of the nonzero features in $\mathbf{X}$ to be preserved during the diffusion. Cell-cell FP using $\overline{\mathbf{A}}^{cell(1)}$ is performed as follows,

$$\overline{\mathbf{X}}^{(1)} = \text{FP}(\mathbf{X}, \overline{\mathbf{A}}^{cell(1)}, \mathbf{M}^{\mathbf{X}}) \tag{1}$$

where $\overline{\mathbf{X}}^{(1)} \in \mathbb{R}^{C \times G}$ is an output of the pre-imputation stage, which is utilized in the following complete relationship stage.

4.3 Complete Relationship Stage

In the complete relationship stage, we refine $\overline{\mathbf{X}}^{(1)}$ through gene-gene and cell-cell propagation. ScBFP [12] adopts gene-gene FP on the k-NN graph constructed based on cosine similarity. However, it overlooks two key issues: **(1)** The similarity-based gene-gene k-NN graph can connect only associating (or co-occurring) genes, excluding highly correlated dissociating (or avoiding) genes, which can offer important biological information for imputation. This occurs because associating genes may have high cosine similarity due to their co-occurrence. These genes with high cosine similarity may become connected in the cosine-similarity-based k-NN graph . **(2)** As shown in Figure 3, since each gene has the distinct distribution within a gene expression matrix [15], the value distribution for a gene varies significantly among genes. Although imputation methods for scRNA-seq generally normalize a gene expression matrix in a cell-wise manner, varying scales across genes still remain after the cell-wise normalization. Therefore, propagation across genes may degrade accurate recovery by mixing values with different scales.

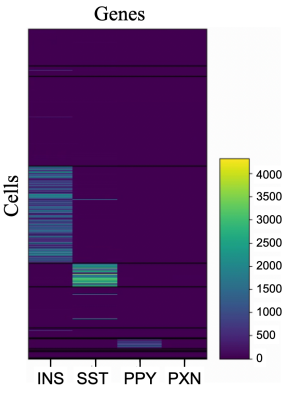

Figure 3: A subset of the gene expression matrix in the Baron Human dataset.

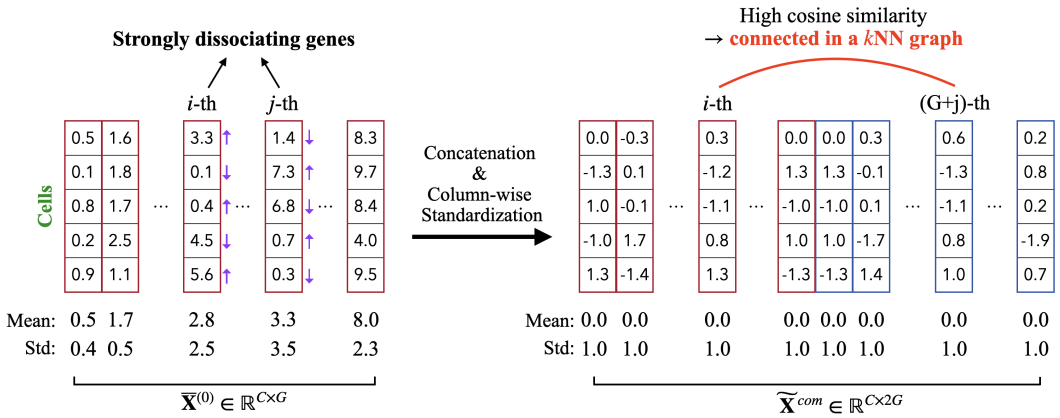

Figure 4: An illustration of the concatenation and standardization processes in the complete relationship stage. Std denotes standard deviation.

Concatenation. To address these key issues, we propose a novel propagation scheme called gene-gene standardized FP. The gene-gene standardized FP first produces $\mathbf{X}^{com} \in \mathbb{R}^{C\times 2G}$ by concatenating $\overline{\mathbf{X}}^{(1)}$ and its negative matrix along columns, *i.e.*, $\mathbf{X}^{com} = [\overline{\mathbf{X}}^{(1)}, -\overline{\mathbf{X}}^{(1)}]$.

Standardization. Subsequently, to enable every gene to have the same scale during propagation among genes, we standardize $\mathbf{X}^{com}$ in a column-wise manner. For $b \in \{1, \ldots, 2G\}$, we standardize $\mathbf{X}^{com}$ to $\widetilde{\mathbf{X}}^{com} \in \mathbb{R}^{C\times 2G}$ as follows,

$$\widetilde{\mathbf{X}}^{com}_{a,b} = \frac{(\mathbf{X}^{com}_{a,b} - \mu_b)}{\sigma_b} \text{ where } \mu_b = \sum_{a=1}^{C} \mathbf{X}^{com}_{a,b},\ \sigma_b = \sqrt{\frac{1}{C-1}\sum_{a=1}^{C}(\mathbf{X}^{com}_{a,b} - \mu_b)^2}. \tag{2}$$

Here, μ_b and σ_b denote the mean and standard deviation of the b-th column (*i.e.*, gene) of $\mathbf{X}^{com}$. Since all the columns in $\widetilde{\mathbf{X}}^{com}$ are standardized, SFP can effectively perform propagation-based imputation without the mixing of values at various scales, addressing the aforementioned issue **(2)**. Furthermore, by concatenating $\overline{\mathbf{X}}^{(1)}$ with its negative matrix before the construction of $\widetilde{\mathbf{X}}^{com}$, SFP can connect not only associating but also dissociating gene-gene relationships. As demonstrated in Figure 4, assume that the i-th gene and the j-th gene have strong dissociating relationships, where $i, j \in \{1, \ldots, G\}$. Within any a-th cell ($a \in \{1, \ldots, C\}$), $\overline{\mathbf{X}}^{(1)}_{a,i}$ will be very high when $\overline{\mathbf{X}}^{(1)}_{a,j}$ is very low and vice versa. After the standardization, the cosine similarity between the i-th gene and the j-th gene will have a large negative value, which cannot be connected in a k-NN graph. However, through the concatenation, $\widetilde{\mathbf{X}}^{com}_{:,(G+j)}$ corresponds to $-\overline{\mathbf{X}}^{(1)}_{:,j}$. Thus, the cosine similarity between $\widetilde{\mathbf{X}}^{com}_{:,i}$ and $\widetilde{\mathbf{X}}^{com}_{:,(G+j)}$ has a large positive value, which will be connected in a k-NN graph. Therefore, through gene-gene propagation using this k-NN graph constructed by using $\widetilde{\mathbf{X}}^{com}$, we can effectively exploit complete gene-gene relationships, including associating and dissociating relationships.

Gene-gene FP. We build a gene-gene k-NN graph on $(\widetilde{\mathbf{X}}^{com})^\top$ by $\overline{\mathbf{A}}^{gene} = k\text{NN}((\widetilde{\mathbf{X}}^{com})^\top)$ where $\overline{\mathbf{A}}^{gene} \in \mathbb{R}^{2G\times 2G}$. We then perform the gene-gene standardized FP using $\overline{\mathbf{A}}^{gene}$ as follows:

$$\overline{\mathbf{X}}^{com} = (\text{FP}((\widetilde{\mathbf{X}}^{com})^\top, \mathbf{A}^{gene}, [\mathbf{M}^{\mathbf{X}}, \mathbf{M}^{\mathbf{X}}]^\top))^\top \tag{3}$$

where $\overline{\mathbf{X}}^{com} \in \mathbb{R}^{C\times 2G}$. Since each gene in $\overline{\mathbf{X}}^{com}$ does not have its original scale due to the standardization, we return all the columns in $\overline{\mathbf{X}}^{com}$ to their original scale as follows:

$$\check{\mathbf{X}}^{com}_{a,b} = \sigma_b \overline{\mathbf{X}}^{com}_{a,b} + \mu_b \tag{4}$$

where $\check{\mathbf{X}}^{com} \in \mathbb{R}^{C\times 2G}$ is the rescaled matrix. We then reduce $\check{\mathbf{X}}^{com}$ by retaining the first G columns, and we denote this reduced matrix as $\overline{\mathbf{X}}^{*} \in \mathbb{R}^{C\times G}$. $\overline{\mathbf{X}}^{*}$ is a final output of the gene-gene standardized FP.

Cell-cell FP. $\overline{\mathbf{X}}^{*}$ containing information from complete gene-gene relationships plays a crucial role in scCR by contributing the formation of all subsequent cell-gene matrices. To perform additional

Table 1: Performance on cell clustering, measured by ARI, NMI, and CA. Standard deviation errors are given. Figures highlighted in green indicate performance improvements over the most competitive baseline at each setting.

Dataset	Baron Mouse			Pancreas			Mouse Bladder		
Method	ARI	NMI	CA	ARI	NMI	CA	ARI	NMI	CA
scTAG	0.565±0.016	0.689±0.023	0.526±0.163	0.678±0.141	0.789±0.011	0.69±0.108	0.604±0.149	0.734±0.047	0.605±0.011
DCA	0.447±0.022	0.710±0.010	0.562±0.002	0.566±0.002	0.786±0.001	0.727±0.002	0.447±0.022	0.710±0.010	0.562±0.002
AutoClass	0.408±0.002	0.699±0.002	0.525±0.004	0.564±0.020	0.795±0.009	0.724±0.030	0.506±0.02	0.732±0.009	0.613±0.029
scGNN 2.0	0.441±0.021	0.734±0.029	0.575±0.019	0.562±0.054	0.793±0.049	0.728±0.061	0.488±0.041	0.717±0.015	0.595±0.033
scGCL	0.478±0.001	0.720±0.000	0.645±0.003	0.645±0.061	0.755±0.042	0.747±0.026	0.529±0.002	0.725±0.005	0.598±0.008
MAGIC	0.419±0.007	0.712±0.007	0.557±0.015	0.595±0.007	0.803±0.004	0.765±0.022	0.565±0.004	0.754±0.001	0.651±0.005
scFP	0.613±0.000	0.817±0.000	0.763±0.000	0.802±0.001	0.872±0.000	0.878±0.000	0.655±0.002	0.767±0.000	0.730±0.001
scBFP	0.660±0.000	0.813±0.001	0.763±0.001	**0.864**±0.000	**0.900**±0.001	**0.918**±0.006	0.694±0.000	0.761±0.002	**0.779**±0.001
scCR (Ours)	**0.827**±0.139 (+**25.3**%)	**0.847**±0.034 (+**3.7**%)	**0.846**±0.084 (+**10.9**%)	0.812±0.000 -	0.855±0.000 -	0.873±0.000 -	**0.704**±0.000 (+**1.7**%)	**0.778**±0.000 (+**1.4**%)	0.765±0.000 -

Dataset	Zeisel			Worm Neuron			Baron Human		
Method	ARI	NMI	CA	ARI	NMI	CA	ARI	NMI	CA
scTAG	0.723±0.010	0.716±0.013	0.712±0.029	0.532±0.134	0.641±0.007	0.439±0.003	0.612±0.029	0.718±0.028	0.610±0.158
DCA	0.693±0.005	0.739±0.005	0.764±0.004	0.502±0.017	0.690±0.016	0.700±0.031	0.545±0.001	0.763±0.004	0.558±0.001
AutoClass	0.673±0.006	0.714±0.009	0.746±0.008	0.488±0.002	0.668±0.001	0.699±0.001	0.523±0.02	0.743±0.005	0.553±0.023
scGNN 2.0	0.533±0.050	0.657±0.063	0.666±0.041	0.453±0.061	0.637±0.03	0.653±0.051	0.525±0.031	0.744±0.025	0.569±0.014
scGCL	0.663±0.003	0.715±0.116	0.717±0.001	0.601±0.014	0.676±0.005	0.754±0.012	0.593±0.027	0.744±0.056	0.671±0.077
MAGIC	0.696±0.003	0.747±0.002	0.765±0.002	0.512±0.016	0.719±0.009	0.770±0.013	0.562±0.012	0.788±0.007	0.596±0.012
scFP	0.848±0.000	0.812±0.000	0.886±0.000	0.524±0.330	**0.731**±0.014	0.766±0.031	0.676±0.000	0.826±0.000	0.732±0.000
scBFP	0.835±0.000	0.792±0.000	0.869±0.000	**0.608**±0.000	0.715±0.000	**0.792**±0.000	0.677±0.000	0.827±0.000	0.733±0.000
scCR (Ours)	**0.902**±0.000 (+**6.4**%)	**0.863**±0.000 (+**6.3**%)	**0.952**±0.000 (+**7.5**%)	0.520±0.014 -	0.711±0.006 -	0.746±0.012 -	**0.823**±0.000 (+**21.6**%)	**0.858**±0.000 (+**3.8**%)	**0.827**±0.000 (+**12.8**%)

cell-cell FP using complete gene-gene relationships inherent in $\overline{\mathbf{X}}^{*}$, we construct cell-cell a k-NN graph by $\overline{\mathbf{A}}^{cell(2)} = k\text{NN}(\mathbf{X}^{*})$. We perform cell-cell FP using $\mathbf{X}$ and $\overline{\mathbf{A}}^{cell(2)}$ as follows,

$$\overline{\mathbf{X}}^{(2)} = \text{FP}(\mathbf{X}, \overline{\mathbf{A}}^{cell(2)}, \mathbf{M}^{\mathbf{X}}) \tag{5}$$

where $\overline{\mathbf{X}}^{(2)}$ is an output of this cell-cell FP.

Weighted sum. $\overline{\mathbf{X}}^{(3)}$, which is a final output of complete relationship stage is produced by the weighted sum of $\overline{\mathbf{X}}^{(2)}$ and $\overline{\mathbf{X}}^{(1)}$ as follows:

$$\overline{\mathbf{X}}^{(3)} = \alpha\overline{\mathbf{X}}^{(1)} + (1-\alpha)\overline{\mathbf{X}}^{(2)} \tag{6}$$

where $0 < \alpha < 1$ is a hyperparameter. $\overline{\mathbf{X}}^{(3)}$ can incorporate valuable biological information since complete gene-gene relationships are delivered by $\overline{\mathbf{X}}^{(2)}$.

4.4 Denoising Stage

Cell-cell Soft FP. While the pre-imputation and complete relationship stages focus on imputing zero values, the denoising stage aims to remove noise in the overall values of $\mathbf{X}$ via propagation-based smoothing. To exploit complete gene-gene relationships, the denoising stage utilizes $\overline{\mathbf{X}}^{(3)}$ containing them. We first build a cell-cell k-NN graph by $\overline{\mathbf{A}}^{cell(3)} = k\text{NN}(\overline{\mathbf{X}}^{(3)})$. To denoise $\mathbf{X}$, we adopt Soft FP [11] that does not maintain zero values during propagation. We apply Soft FP [11] to $\mathbf{X}$ as follows,

$$\overline{\mathbf{X}}^{(4)}(t) = \beta\overline{\mathbf{A}}^{cell(3)}\overline{\mathbf{X}}^{(4)}(t-1) + (1-\beta)\mathbf{X}, \;\; t = 1, \cdots, K, \tag{7}$$

where K is the total number of propagation steps, $\overline{\mathbf{X}}^{(4)}(0) = \mathbf{X}$, $\overline{\mathbf{X}}^{(4)}(t) \in \mathbb{R}^{C\times G}$ is the updated cell-gene matrix after t propagation steps, and $0 < \beta < 1$ is a hyperparameter. An output of the denoising stage, denoted by $\overline{\mathbf{X}}^{(4)}(K)$, is obtained after K steps.

Weighted sum. The final output of our scCR, $\widehat{\mathbf{X}}$, is the weighted sum of $\overline{\mathbf{X}}^{(3)}$ and $\overline{\mathbf{X}}^{(4)}(K)$ as follows:

$$\widehat{\mathbf{X}} = \gamma\overline{\mathbf{X}}^{(3)} + (1-\gamma)\overline{\mathbf{X}}^{(4)}(K). \tag{8}$$

where and $0 < \gamma \leq 1$ is a hyperparameter. In summary, unlike existing propagation-based methods, our scCR enables the use of complete gene-gene relationships in denoising scRNA-seq.

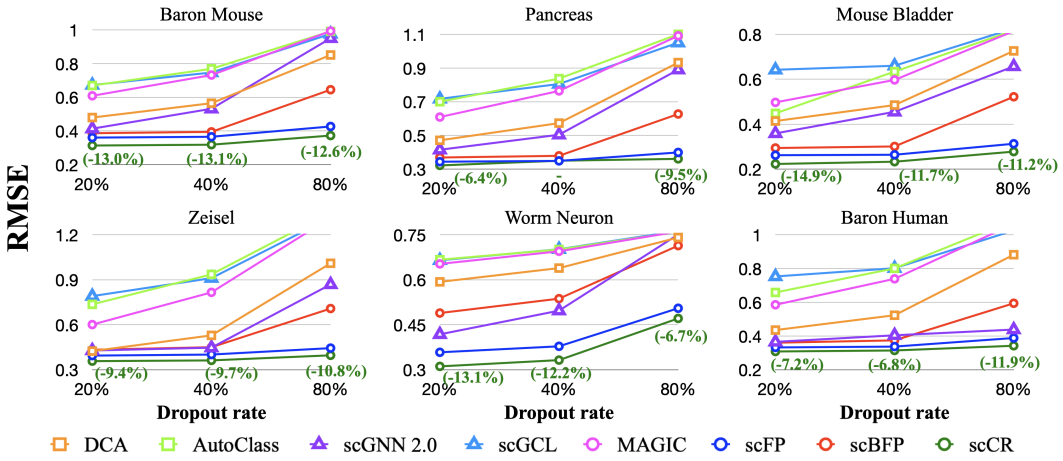

Figure 5: Performance on dropout recovery, measured by RMSE. Figures highlighted in green indicate reduction rates from the most competitive baseline at each setting.

5 Experiments

5.1 Experimental Setup

We performed comparative evaluation of scCR on six widely used scRNA-seq datasets with gold-standard cell type information: Baron Mouse [23], Pancreas [24], Mouse Bladder [25], Zeisel [2], Worm Neuron [26], and Baron Human [23]. We compared our scCR with eight state-of-the-art methods handling noise in scRNA-seq data: (1) non-graph-based methods: DCA [17] and AutoClass [18]; (2) GNN-based methods: scTAG [21], scGNN 2.0 [19], and scGCL [20]; (3) propagation-based methods: MAGIC [22], scFP [11], and scBFP [12]. We evaluated scTAG only on cell clustering, since scTAG is a clustering method. To evaluate the cell clustering of scCR and baselines, we utilized three standard evaluation metrics: Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), and Clustering Accuracy (CA). For dropout recovery, we employed two standard evaluation metrics: Root Mean Square Error (RMSE) and Median L1 Distance.

Implementation. In all experiments, the hyperparmeters of scCR are set to default settings, taking into account the unsupervised nature of the single-cell analysis. For fair comparisons, we set the hyperparameters of baselines according to specifications in the papers and official codes. We reported the average performance across three independent runs. Experimental details regarding datasets, baselines, evaluation metrics, and hyperparameter settings are provided in Appendix G.

5.2 Results

scCR enables improved cell clustering. To validate the effectiveness of scCR in cell clustering, we evaluated its clustering performance. Table 1 presents the performance comparison. While FP-based baselines, including scFP and scBFP, outperformed other baselines, scCR delivered the best or competitive cell clustering performance across all datasets. Specifically, scCR improved ARI by 25.3%, 1.7%, 6.4%, and 21.6% compared to previous state-of-the-art results on Baron Mouse, Mouse Bladder, Zeisel, and Baron Human, respectively.

scCR effectively recovers dropout values. Since dropouts can occur at various rates, we generated false zeros (*i.e.*, dropouts) at non-zero values in datasets by applying varying dropout rates. As shown in Figure 5, scCR significantly improved dropout recovery performance in various dropout rates across the datasets. We confirmed that scCR effectively reduced RMSE between imputed values and their original values with large reduction rates, only except for the Pancreas dataset with 40% dropout. We provided a recovery performance comparison, measured by the median L1 distance in Appendix H.2, which also demonstrates the effectiveness of scCR.

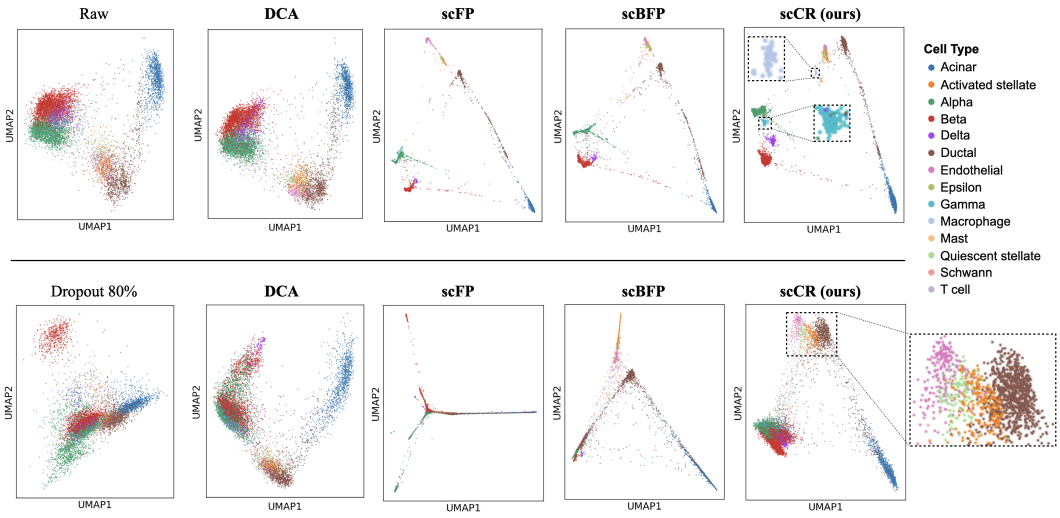

Figure 6: UMAP visualization using the Baron Human dataset, comparing scCR with the three most competitive imputation methods. The first row shows the visualization of the raw data and their imputed results. The second row displays the visualization of data subjected to an 80% dropout rate and their imputed results.

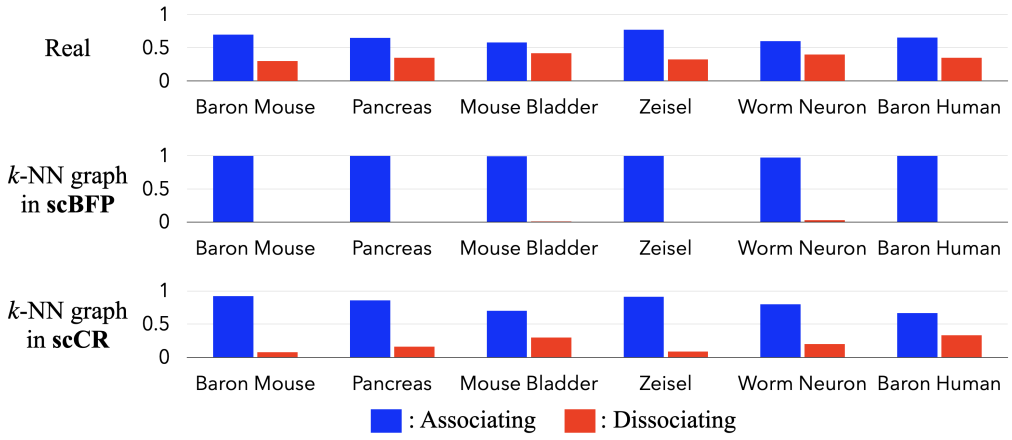

Figure 7: The first row indicates the percentages of associating and dissociating gene-gene relationships in datasets. The second and third rows represents the percentages of associating and dissociating relationships within a gene-gene k-NN graph in each method.

scCR identifies rare cell types well. To verify the effectiveness of scCR in identifying rare cell types, we conducted in-depth analysis using the Baron Human dataset. We visualized the two-dimensional UMAP [27] repesentations of the raw data and the data with an 80% dropout rate applied. As shown in the first row in Figure 6, scCR effectively identified rare cell types with few cells, such as 'gamma' and 'Macrophage'. It is noteworthy that even under severe dropouts, scCR separated cell types well (in the second row), whereas compared methods failed.

Does scCR really model dissociating relationships? To show that scCR models both associating and dissociating gene-gene relationships, we first investigated the ratio of dissociating relationships to associating relationships. For this investigation, we define that associating or dissociating relationships exist when the absolute values of the correlation coefficients between genes exceed the top 25%. We determine that a positive sign in the correlation coefficients indicates associating relationships, while a negative sign indicates dissociating relationships. We then measured the percentages of

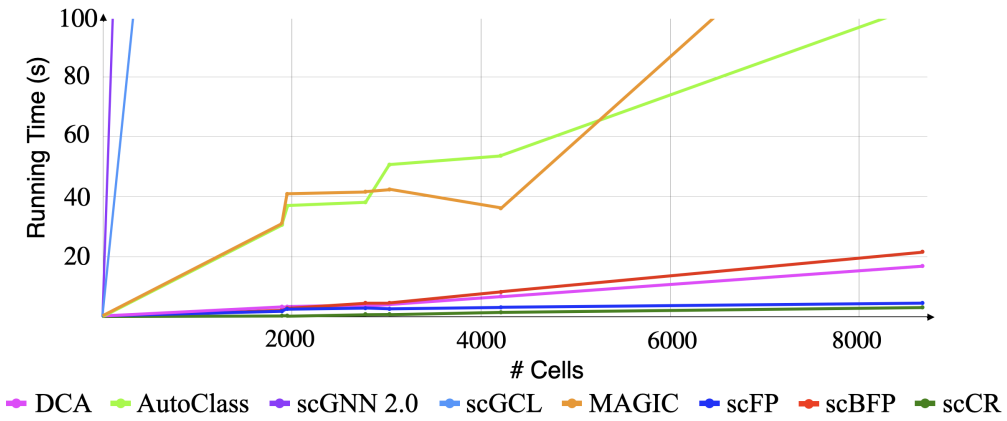

Figure 8: Running time comparison of scCR and baselines according to the number of cells.

associating and dissociating relationships within gene-gene k-NN graphs in scBFP and our scCR. As shown in Figure 7, scBFP hardly models dissociating gene-gene relationships. In contrast, scCR effectively models dissociating relationships, enabling the use of complete gene-gene relationships in its imputation.

scCR is even faster than existing imputation methods. To show the advantage of scCR in terms of imputation time, we measured the running time of scCR and imputation methods on datasets. As shown in Figure 8, scCR showed the lowest running time across all the datasets, regardless of the number of cells.

An ablation study (See Appendix C), further experimental results (See Appendix H), and the proof of convergence of FP (See Appendix B) are provided in Appendix.

6 Conclusion

In this paper, we proposed a novel imputation framework called Single-Cell Complete Relationship (scCR) for scRNA-seq data imputation. scCR utilized complete gene-gene relationships by concatenating a given cell-gene matrix with its negation and facilitated effective gene-gene propagation through the standardization of the cell-gene matrix in a gene-wise manner. These processes, grounded in genetic evidence, led to significant performance improvements over state-of-the-art methods in various downstream tasks on scRNA-seq data, with fast imputation times. Furthermore, our work is not limited to simply utilizing genetic evidence to design a framework. We validated this evidence within real scRNA-seq datasets, and confirmed that our scCR effectively leveraged this insight. Like other scRNA-seq imputation methods, scCR is specifically designed for scRNA-seq data. However, since scRNA-seq data are inherently matrix-formatted, scCR can be extended to general tabular data imputation. We expect that the concatenation process will be effective even for general tabular data, as there are often both positive and negative correlation coefficients between channels in such data. The extension of scCR to other domains is left for future work.

Broader Impacts

Our work provides an important insight that, when applying machine learning to the biomedical domain, it is crucial to approach with biological grounding rather than focusing solely on applying existing cutting-edge machine learning techniques. Since scRNA-seq has opened a new frontier for understanding biological systems [15, 28, 29], we believe that our work will contribute to the biomedical domains by enhancing the analysis of human diseases and the discovery of new genetic observations. We have not identified any negative impacts of our work on society.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (2021-0-01341,Artificial Intelligence Graduate School Program(Chung-Ang University)).

References

[1] Evan Z Macosko, Anindita Basu, Rahul Satija, James Nemesh, Karthik Shekhar, Melissa Goldman, Itay Tirosh, Allison R Bialas, Nolan Kamitaki, Emily M Martersteck, et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161 (5):1202–1214, 2015.

[2] Amit Zeisel, Ana B Muñoz-Manchado, Simone Codeluppi, Peter Lönnerberg, Gioele La Manno, Anna Juréus, Sueli Marques, Hermany Munguba, Liqun He, Christer Betsholtz, et al. Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq. *Science*, 347(6226): 1138–1142, 2015.

[3] Michael JT Stubbington, Orit Rozenblatt-Rosen, Aviv Regev, and Sarah A Teichmann. Single-cell transcriptomics to explore the immune system in health and disease. *Science*, 358(6359): 58–63, 2017.

[4] Hadas Keren-Shaul, Amit Spinrad, Assaf Weiner, Orit Matcovitch-Natan, Raz Dvir-Szternfeld, Tyler K Ulland, Eyal David, Kuti Baruch, David Lara-Astaiso, Beata Toth, et al. A unique microglia type associated with restricting development of alzheimer's disease. *Cell*, 169(7): 1276–1290, 2017.

[5] Cole Trapnell, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J Lennon, Kenneth J Livak, Tarjei S Mikkelsen, and John L Rinn. Pseudo-temporal ordering of individual cells reveals dynamics and regulators of cell fate decisions. *Nature biotechnology*, 32(4):381, 2014.

[6] Monika M Gladka, Bas Molenaar, Hesther De Ruiter, Stefan Van Der Elst, Hoyee Tsui, Danielle Versteeg, Grègory PA Lacraz, Manon MH Huibers, Alexander Van Oudenaarden, and Eva Van Rooij. Single-cell sequencing of the healthy and diseased heart reveals cytoskeleton-associated protein 4 as a new modulator of fibroblasts activation. *Circulation*, 138(2):166–180, 2018.

[7] William Stephenson, Laura T Donlin, Andrew Butler, Cristina Rozo, Bernadette Bracken, Ali Rashidfarrokhi, Susan M Goodman, Lionel B Ivashkiv, Vivian P Bykerk, Dana E Orange, et al. Single-cell rna-seq of rheumatoid arthritis synovial tissue using low-cost microfluidic instrumentation. *Nature communications*, 9(1):791, 2018.

[8] Stephanie C Hicks, F William Townes, Mingxiang Teng, and Rafael A Irizarry. Missing data and technical variability in single-cell rna-sequencing experiments. *Biostatistics*, 19(4):562–578, 2018.

[9] Florian Buettner, Kedar N Natarajan, F Paolo Casale, Valentina Proserpio, Antonio Scialdone, Fabian J Theis, Sarah A Teichmann, John C Marioni, and Oliver Stegle. Computational analysis of cell-to-cell heterogeneity in single-cell rna-sequencing data reveals hidden subpopulations of cells. *Nature biotechnology*, 33(2):155–160, 2015.

[10] Uri Shaham, Kelly P Stanton, Jun Zhao, Huamin Li, Khadir Raddassi, Ruth Montgomery, and Yuval Kluger. Removal of batch effects using distribution-matching residual networks. *Bioinformatics*, 33(16):2539–2546, 2017.

[11] Sukwon Yun, Junseok Lee, and Chanyoung Park. Single-cell rna-seq data imputation using feature propagation. *arXiv preprint arXiv:2307.10037*, 2023.

[12] Junseok Lee, Sukwon Yun, Yeongmin Kim, Tianlong Chen, Manolis Kellis, and Chanyoung Park. Single-cell rna sequencing data imputation using bi-level feature propagation. *Briefings in Bioinformatics*, 25(3):bbae209, 2024.

[13] Fiona Jane Whelan, Martin Rusilowicz, and James Oscar McInerney. Coinfinder: detecting significant associations and dissociations in pangenomes. *Microbial genomics*, 6(3):e000338, 2020.

[14] Rebecca J Hall, Fiona J Whelan, Elizabeth A Cummins, Christopher Connor, Alan McNally, and James O McInerney. Gene-gene relationships in an escherichia coli accessory genome are linked to function and mobility. *Microbial Genomics*, 7(9):000650, 2021.

[15] Anoop P Patel, Itay Tirosh, John J Trombetta, Alex K Shalek, Shawn M Gillespie, Hiroaki Wakimoto, Daniel P Cahill, Brian V Nahed, William T Curry, Robert L Martuza, et al. Single-cell rna-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*, 344(6190): 1396–1401, 2014.

[16] Wei Vivian Li and Jingyi Jessica Li. An accurate and robust imputation method scimpute for single-cell rna-seq data. *Nature communications*, 9(1):997, 2018.

[17] Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):390, 2019.

[18] Hui Li, Cory R Brouwer, and Weijun Luo. A universal deep neural network for in-depth cleaning of single-cell rna-seq data. *Nature Communications*, 13(1):1901, 2022.

[19] Juexin Wang, Anjun Ma, Yuzhou Chang, Jianting Gong, Yuexu Jiang, Ren Qi, Cankun Wang, Hongjun Fu, Qin Ma, and Dong Xu. scgnn is a novel graph neural network framework for single-cell rna-seq analyses. *Nature communications*, 12(1):1882, 2021.

[20] Zehao Xiong, Jiawei Luo, Wanwan Shi, Ying Liu, Zhongyuan Xu, and Bo Wang. scgcl: an imputation method for scrna-seq data based on graph contrastive learning. *Bioinformatics*, 39 (3):btad098, 2023.

[21] Zhuohan Yu, Yifu Lu, Yunhe Wang, Fan Tang, Ka-Chun Wong, and Xiangtao Li. Zinb-based graph embedding autoencoder for single-cell rna-seq interpretations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4671–4679, 2022.

[22] David Van Dijk, Roshan Sharma, Juozas Nainys, Kristina Yim, Pooja Kathail, Ambrose J Carr, Cassandra Burdziak, Kevin R Moon, Christine L Chaffer, Diwakar Pattabiraman, et al. Recovering gene interactions from single-cell data using data diffusion. *Cell*, 174(3):716–729, 2018.

[23] Maayan Baron, Adrian Veres, Samuel L Wolock, Aubrey L Faust, Renaud Gaujoux, Amedeo Vetere, Jennifer Hyoje Ryu, Bridget K Wagner, Shai S Shen-Orr, Allon M Klein, et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. *Cell systems*, 3(4):346–360, 2016.

[24] Malte D Luecken, Maren Büttner, Kridsadakorn Chaichoompu, Anna Danese, Marta Interlandi, Michaela F Müller, Daniel C Strobl, Luke Zappia, Martin Dugas, Maria Colomé-Tatché, et al. Benchmarking atlas-level data integration in single-cell genomics. *Nature methods*, 19(1): 41–50, 2022.

[25] Xiaoping Han, Renying Wang, Yincong Zhou, Lijiang Fei, Huiyu Sun, Shujing Lai, Assieh Saadatpour, Ziming Zhou, Haide Chen, Fang Ye, et al. Mapping the mouse cell atlas by microwell-seq. *Cell*, 172(5):1091–1107, 2018.

[26] Junyue Cao, Jonathan S Packer, Vijay Ramani, Darren A Cusanovich, Chau Huynh, Riza Daza, Xiaojie Qiu, Choli Lee, Scott N Furlan, Frank J Steemers, et al. Comprehensive single-cell transcriptional profiling of a multicellular organism. *Science*, 357(6352):661–667, 2017.

[27] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29), 2018.

[28] Diego Adhemar Jaitin, Ephraim Kenigsberg, Hadas Keren-Shaul, Naama Elefant, Franziska Paul, Irina Zaretsky, Alexander Mildner, Nadav Cohen, Steffen Jung, Amos Tanay, et al. Massively parallel single-cell rna-seq for marker-free decomposition of tissues into cell types. *Science*, 343(6172):776–779, 2014.

[29] Nicholas Schaum, Jim Karkanias, Norma F Neff, Andrew P May, Stephen R Quake, Tony Wyss-Coray, Spyros Darmanis, Joshua Batson, Olga Botvinnik, Michelle B Chen, et al. Single-cell transcriptomics of 20 mouse organs creates a tabula muris: The tabula muris consortium. *Nature*, 562(7727):367, 2018.

[30] 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*, pages 11–1. PMLR, 2022.

[31] Abraham Berman and Robert J Plemmons. *Nonnegative matrices in the mathematical sciences*. SIAM, 1994.

[32] Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

[33] Yunqing Liu, Jiayi Zhao, Taylor S Adams, Ningya Wang, Jonas C Schupp, Weimiao Wu, John E McDonough, Geoffrey L Chupp, Naftali Kaminski, Zuoheng Wang, et al. idesc: identifying differential expression in single-cell rna sequencing data with multiple subjects. *BMC bioinformatics*, 24(1):318, 2023.

[34] F Alexander Wolf, Philipp Angerer, and Fabian J Theis. Scanpy: large-scale single-cell gene expression data analysis. *Genome biology*, 19:1–5, 2018.

A Feature Propagation

Assume that a given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ has a feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$ with missing values, where rows and columns correspond to nodes and F feature channels, respectively. We use $\overline{\mathbf{A}} \in \mathbb{R}^{N \times N}$ to denote a normalized adjacency, which is an input for feature propagation (FP). To preserve known features during the diffusion process, we mark the positions of the features to be preserved with 1 in the mask matrix $\mathbf{M} \in \{0, 1\}^{N \times F}$. Here, values of 1 in $\mathbf{M}$ indicate the location of observed features, where feature values will be preserved.

To formally explain the diffusion process of FP in detail, we temporarily reorder nodes for notational convenience. Since observed values may differ across channels, we reorder the nodes for each channel. When we consider the a-th channel, based on the values of 1 in $\mathbf{M}_{:,a}$, we let $\mathcal{V}_k^a$ be the set of nodes whose a-th values are known (observed). Similarly, by examining zero values in $\mathbf{M}_{:,a}$, $\mathcal{V}_u^a$ denotes the set of nodes whose a-th values are unknown (missing). By reordering the nodes in the order of $\mathcal{V}_k^a$ and $\mathcal{V}_u^a$, the entire feature values and the adjacency matrix for the a-th channel can be expressed as

$$\mathbf{x}^a = \begin{bmatrix} \mathbf{x}_k^a \\ \mathbf{x}_u^a \end{bmatrix}, \qquad \overline{\mathbf{A}}^{(a)} = \begin{bmatrix} \overline{\mathbf{A}}_{kk}^{(a)} & \overline{\mathbf{A}}_{ku}^{(a)} \\ \overline{\mathbf{A}}_{uk}^{(a)} & \overline{\mathbf{A}}_{uu}^{(a)} \end{bmatrix}, \tag{9}$$

where $\mathbf{x}^a$ is a column vector representing features for the a-th channel in $\mathbf{X}$ in the order of $\mathcal{V}_k^a$ and $\mathcal{V}_u^a$. Here, $\mathbf{x}_k^a \in \mathbb{R}^{|\mathcal{V}_k^a|}$ and $\mathbf{x}_u^a \in \mathbb{R}^{|\mathcal{V}_u^a|}$. Similarly, $\overline{\mathbf{A}}^{(a)}$ consists of four sub-matrices related to $\mathcal{V}_k^a$ and $\mathcal{V}_u^a$. It is noteworthy that although $\overline{\mathbf{A}}^{(a)} \in \mathbb{R}^{N \times N}$ and $\overline{\mathbf{A}} \in \mathbb{R}^{N \times N}$ is different due to reordering, they represent the same graph structure.

To preserve observed values during the diffusion process, we replace the first $|\mathcal{V}_k^a|$ rows in $\overline{\mathbf{A}}^{(a)}$ with one-hot vectors indicating $\mathcal{V}_k^a$. Consequently, we obtain a transition matrix $\widetilde{\mathbf{A}}^{(a)} \in \mathbb{R}^{N \times N}$ expressed by

$$\widetilde{\mathbf{A}}^{(a)} = \begin{bmatrix} \mathbf{I}_{kk} & \mathbf{0}_{ku} \\ \overline{\mathbf{A}}_{uk}^{(a)} & \overline{\mathbf{A}}_{uu}^{(a)} \end{bmatrix}, \tag{10}$$

where $\mathbf{I}_{nn} \in \mathbb{R}^{|\mathcal{V}_n^a| \times |\mathcal{V}_n^a|}$ is an identity matrix and $\mathbf{0}_{nz} \in \mathbb{R}^{|\mathcal{V}_n^a| \times |\mathcal{V}_z^a|}$ is a zero matrix. The diffusion process of FP is implemented by iterative propagation steps utilizing $\widetilde{\mathbf{A}}^{(a)}$ as

$$\begin{aligned} \bar{\mathbf{x}}^a(t) &= \widetilde{\mathbf{A}}^{(a)} \bar{\mathbf{x}}^a(t-1), \;\; t = 1, \cdots, K; \\ \bar{\mathbf{x}}^a(0) &= \begin{bmatrix} \mathbf{x}_k^a \\ \mathbf{0}_u^a \end{bmatrix}, \end{aligned} \tag{11}$$

where $\bar{\mathbf{x}}^a(t)$ is an imputed feature vector after t propagation steps and $\mathbf{0}_u^a$ denotes a zero vector of the same length as $|\mathcal{V}_k^a|$. As $K \to \infty$, this recursion converges and $\bar{\mathbf{x}}^a(t)$ reaches a steady state (the proof can be found in Appendix B). We use $\bar{\mathbf{x}}^a(K)$ with large enough K to approximate the steady state.

After this diffusion process for the entire channels, we attain $\{\bar{\mathbf{x}}^a(K)\}_{a=1}^F$. Since these vectors have different ordering due to channel-wise reordering, we rearrange $\{\bar{\mathbf{x}}^a(K)\}_{a=1}^F$ in the original order and construct $\overline{\mathbf{X}} \in \mathbb{R}^{N \times F}$ by stacking the originally ordered vectors in $\{\bar{\mathbf{x}}^a(K)\}_{a=1}^F$ along the channels. In summary, FP fills in missing values in $\mathbf{X}$ through diffusion using $\overline{\mathbf{A}}$ while preserving features corresponding to values of 1 in $\mathbf{M}$.

B Proof of Convergence of Feature Propagation

Feature propagation (FP) [30] utilize symmetrically normalized transition matrix for the diffusion process implemented by iterative propagation steps. We prove the convergence of this diffusion process as follows.

Proposition 1. *The transition matrix for the a-th channel is defined by*

$$\widetilde{\mathbf{A}}^{(a)} = \begin{bmatrix} \mathbf{I}_{kk} & \mathbf{0}_{ku} \\ \overline{\mathbf{A}}_{uk}^{(a)} & \overline{\mathbf{A}}_{uu}^{(a)} \end{bmatrix},$$

where $\widetilde{\mathbf{A}}^{(a)}$ is symmetrically normalized. Using $\widetilde{\mathbf{A}}^{(a)}$, the diffusion process in the a-th channel is defined by

$$\bar{\mathbf{x}}^a(t) = \widetilde{\mathbf{A}}^{(a)}\bar{\mathbf{x}}^a(t-1), \;\; t = 1, \cdots, K;$$
$$\bar{\mathbf{x}}^a(0) = \begin{bmatrix} \mathbf{x}_k^a \\ \mathbf{0}_u^a \end{bmatrix},$$

Then, $\lim_{K\to\infty} \bar{\mathbf{x}}^{(a)}(K)$ converges.

The proof of Proposition 1 refers to [30]. We begin with two lemmas.

Lemma 1. *$\overline{\mathbf{A}}^{(a)}$ is the symmetrically normalized matrix calculated by $\overline{\mathbf{A}}^{(a)} = (\mathbf{D}^{(a)})^{-1/2}\mathbf{A}^{(a)}(\mathbf{D}^{(a)})^{-1/2}$ where $\mathbf{D}^{(a)}$ is a diagonal matrix that has diagonal entities $\mathbf{D}_{ii}^{(a)} = \sum_j \mathbf{A}_{i,j}^{(a)}$. $\overline{\mathbf{A}}_{uu}^{(a)}$ is the $|\bar{\mathbf{x}}_u^{(a)}| \times |\bar{\mathbf{x}}_u^{(a)}|$ bottom-right submatrix of $\overline{\mathbf{A}}^{(a)}$ and let $\rho(\cdot)$ denote spectral radius. Then, $\rho(\overline{\mathbf{A}}_{uu}^{(a)}) < 1$.*

Proof. Consider $\overline{\mathbf{A}}_{uu0}^{(a)} \in \mathbb{R}^{N\times N}$, where the bottom right submatrix is equal to $\overline{\mathbf{A}}_{uu}^{(a)}$ and all other elements are zero. That is,

$$\overline{\mathbf{A}}_{uu0}^{(a)} = \begin{bmatrix} \mathbf{0}_{kk} & \mathbf{0}_{ku} \\ \mathbf{0}_{uk} & \overline{\mathbf{A}}_{uu}^{(a)} \end{bmatrix}$$

where $\mathbf{0}_{kk} \in \{0\}^{|\bar{\mathbf{x}}_k^{(a)}|\times|\bar{\mathbf{x}}_k^{(a)}|}$, $\mathbf{0}_{ku} \in \{0\}^{|\bar{\mathbf{x}}_k^{(a)}|\times|\bar{\mathbf{x}}_u^{(a)}|}$, and $\mathbf{0}_{uk} \in \{0\}^{|\bar{\mathbf{x}}_u^{(a)}|\times|\bar{\mathbf{x}}_k^{(a)}|}$. Given that $\overline{\mathbf{A}}^{(a)}$ represents the weighted adjacency matrix of the connected graph $\mathcal{G}$, $\overline{\mathbf{A}}_{uu0}^{(a)} \leq \overline{\mathbf{A}}^{(a)}$ element-wise and $\overline{\mathbf{A}}_{uu0}^{(a)} \neq \overline{\mathbf{A}}^{(a)}$. Furthermore, considering that $\overline{\mathbf{A}}_{uu0}^{(a)} + \overline{\mathbf{A}}^{(a)}$ constitutes the weighted adjacency matrix of a strongly connected graph, we can conclude that $\overline{\mathbf{A}}_{uu0}^{(a)} + \overline{\mathbf{A}}^{(a)}$ is irreducible based on Theorem 2.2.7 in [31]. Consequently, applying Corollary 2.1.5 in [31], $\rho(\overline{\mathbf{A}}_{uu0}^{(a)}) < \rho(\overline{\mathbf{A}}^{(a)})$. Furthermore, $\rho(\overline{\mathbf{A}}^{(a)}) \leq 1$ since we can write $\overline{\mathbf{A}}^{(a)} = \mathbf{I} - (\mathbf{D}^{(a)})^{-1/2}\mathbf{A}(\mathbf{D}^{(a)})^{-1/2}$, where $(\mathbf{D}^{(a)})^{-1/2}\mathbf{A}(\mathbf{D}^{(a)})^{-1/2}$ has eigenvalues in the range $[0, 2]$ [32]. Note that since both $\overline{\mathbf{A}}_{uu0}^{(a)}$ and $\overline{\mathbf{A}}_{uu}^{(a)}$ share the same non-zero eigenvalues, it follows that $\rho(\overline{\mathbf{A}}_{uu0}^{(a)}) = \rho(\overline{\mathbf{A}}_{uu}^{(a)})$. Ultimately, combining these inequalities leads to the result $\rho(\overline{\mathbf{A}}_{uu}^{(a)}) = \rho(\overline{\mathbf{A}}_{uu0}^{(a)}) < \rho(\overline{\mathbf{A}}^{(a)}) = 1$. □

Lemma 2. *$\mathbf{I}_{uu} - \overline{\mathbf{A}}_{uu}^{(a)}$ is invertible where $\mathbf{I}_{uu}$ is the $|\bar{\mathbf{x}}_u^{(a)}| \times |\bar{\mathbf{x}}_u^{(a)}|$ identity matrix.*

Proof. Since 1 is not an eigenvalue of $\overline{\mathbf{A}}_{uu}^{(a)}$ by Lemma 1, 0 is not an eigenvlaue of $\mathbf{I}_{uu} - \overline{\mathbf{A}}_{uu}^{(a)}$. Thus $\mathbf{I}_{uu} - \overline{\mathbf{A}}_{uu}^{(a)}$ is invertible. □

We now prove Propostion 1 as follows.

Proof. The recursive relation can be written as

$$\bar{\mathbf{x}}^{(a)}(t) = \begin{bmatrix} \bar{\mathbf{x}}_k^{(a)}(t) \\ \bar{\mathbf{x}}_u^{(a)}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{kk} & \mathbf{0}_{ku} \\ \overline{\mathbf{A}}_{uk}^{(a)} & \overline{\mathbf{A}}_{uu}^{(a)} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{x}}_k^{(a)}(t-1) \\ \bar{\mathbf{x}}_u^{(a)}(t-1) \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{x}}_k^{(a)}(t-1) \\ \overline{\mathbf{A}}_{uk}^{(a)}\bar{\mathbf{x}}_k^{(a)}(t-1) + \overline{\mathbf{A}}_{uu}^{(a)}\bar{\mathbf{x}}_u^{(a)}(t-1) \end{bmatrix}.$$

Since $\bar{\mathbf{x}}_k^{(a)}(t) = \bar{\mathbf{x}}_k^{(a)}(t-1)$ in the first $|\bar{\mathbf{x}}_k^{(a)}|$ rows, it follows that $\bar{\mathbf{x}}_k^{(a)}(K) = \ldots = \bar{\mathbf{x}}_k^{(a)}$. That is, $\bar{\mathbf{x}}_k^{(a)}(K)$ retains the values of $\mathbf{x}_k^{(a)}$. Therefore, $\lim_{K\to\infty} \bar{\mathbf{x}}_k^{(a)}(K)$ converges to $\mathbf{x}_k^{(a)}$.

Table 2: Ablation study. Con and Sta denote the concatenation and standardization, respectively, which enable capturing dissociating gene-gene relationships.

Con	Sta	Baron Mouse			Zeisel			Baron Human		
		ARI	NMI	CA	ARI	NMI	CA	ARI	NMI	CA
✗	✗	$0.625_{\pm 0.001}$	$0.777_{\pm 0.001}$	$0.719_{\pm 0.003}$	$0.789_{\pm 0.000}$	$0.738_{\pm 0.000}$	$0.851_{\pm 0.000}$	$0.805_{\pm 0.000}$	$0.837_{\pm 0.000}$	$0.813_{\pm 0.000}$
✗	✓	$0.623_{\pm 0.000}$	$0.788_{\pm 0.001}$	$0.710_{\pm 0.001}$	$0.898_{\pm 0.000}$	$0.856_{\pm 0.000}$	$0.948_{\pm 0.000}$	$0.816_{\pm 0.000}$	$0.846_{\pm 0.002}$	$0.819_{\pm 0.001}$
✓	✗	$0.629_{\pm 0.001}$	$0.791_{\pm 0.000}$	$0.731_{\pm 0.002}$	$0.821_{\pm 0.000}$	$0.794_{\pm 0.000}$	$0.839_{\pm 0.000}$	$0.808_{\pm 0.002}$	$0.841_{\pm 0.001}$	$0.818_{\pm 0.002}$
✓	✓	$\mathbf{0.827}_{\pm 0.093}$	$\mathbf{0.847}_{\pm 0.034}$	$\mathbf{0.846}_{\pm 0.084}$	$\mathbf{0.902}_{\pm 0.000}$	$\mathbf{0.863}_{\pm 0.000}$	$\mathbf{0.952}_{\pm 0.000}$	$\mathbf{0.823}_{\pm 0.000}$	$\mathbf{0.858}_{\pm 0.000}$	$0.827_{\pm 0.000}$

Table 3: Further ablation study of scCR on cell clustering measured by ARI. PRE, COM, and DEN denote the pre-imputation stage, the complete relation stage, and the denosing stage, respectively.

PRE	COM	DEN	Baron Mouse	Zeisel	Baron Human
✓	✗	✗	0.437 ± 0.061	0.682 ± 0.024	0.580 ± 0.036
✓	✓	✗	0.409 ± 0.008	0.732 ± 0.001	0.571 ± 0.007
✓	✗	✓	0.584 ± 0.000	0.822 ± 0.000	0.681 ± 0.000
✓	✓	✓	$\mathbf{0.827 \pm 0.093}$	$\mathbf{0.902 \pm 0.000}$	$\mathbf{0.823 \pm 0.000}$

We now focus solely on the convergence of $\lim_{K\to\infty} \bar{\mathbf{x}}_u^{(a)}(K)$. When we unroll the recursion for the last $|\bar{\mathbf{x}}_u^{(a)}|$ rows,

$$\begin{aligned}
\bar{\mathbf{x}}_u^{(a)}(K) &= \overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)} + \overline{\mathbf{A}}_{uu}^{(a)}\bar{\mathbf{x}}_u^{(a)}(K-1) \\
&= \overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)} + \overline{\mathbf{A}}_{uu}^{(a)}(\overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)} + \overline{\mathbf{A}}_{uu}^{(a)}\bar{\mathbf{x}}_u^{(a)}(K-2)) \\
&= \dots \\
&= (\sum_{t=0}^{K-1}(\overline{\mathbf{A}}_{uu}^{(a)})^t)\overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)} + (\overline{\mathbf{A}}_{uu}^{(a)})^K\bar{\mathbf{x}}_u^{(a)}(0)
\end{aligned}$$

By Lemma 1, $\lim_{K\to\infty}(\overline{\mathbf{A}}_{uu}^{(a)})^K = 0$. Therefore, $\lim_{K\to\infty}(\overline{\mathbf{A}}_{uu}^{(a)})^K\bar{\mathbf{x}}_u^{(a)}(0) = 0$, regardless of the initial state for $\bar{\mathbf{x}}_u^{(a)}(0)$. (we replace $\bar{\mathbf{x}}_u^{(a)}(0)$ with a zero column vector for simplicity.) Hence, our focus shifts to $\lim_{K\to\infty}(\sum_{t=0}^{K-1}(\overline{\mathbf{A}}_{uu}^{(a)})^t)\overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)}$.

Given that Lemma 1 establishes $\rho(\overline{\mathbf{A}}_{uu}^{(a)}) < 1$, and Lemma 2 affirms the invertibility of $(\mathbf{I}_{uu} - \overline{\mathbf{A}}_{uu}^{(a)})^{-1}$, the geometric series converges as follows

$$\lim_{K\to\infty}\bar{\mathbf{x}}_u^{(a)}(K) = \lim_{K\to\infty}(\sum_{t=0}^{K-1}(\overline{\mathbf{A}}_{uu}^{(a)})^t)\overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)} = (\mathbf{I}_{uu} - \overline{\mathbf{A}}_{uu}^{(a)})^{-1}\overline{\mathbf{A}}_{uk}^{(a)}\mathbf{x}_k^{(a)}.$$

In conclusion, the recursion in FP converges. □

C Ablation Study

We conducted an ablation study to analyze the effectiveness of each component in scCR. We performed cell clustering on the Baron Mouse, Zeisel, and Baron Human datasets. As shown in Table 2, both concatenation and standardization contributed to performance improvement, and the combination of the two components led to significant performance improvement.

We conducted an additional ablation study to assess the effectiveness of each stage of scCR. Table 3 presents the results of the ablation study in terms of cell clustering, measured by ARI. As shown in the table, adding the complete relation stage and the denoising stage significantly improved performance compared to using only the pre-imputation stage. These results confirm that the complete relation and denoising stages contributed substantially to the high performance of scCR, underscoring the well-founded design of our approach.

Table 4: Performance on dropout recovery under Missing Not At Random (MNAR) settings, measured by RMSE.

Dataset	scFP	scBFP	scCR (ours)
Baron Mouse	0.517 ± 0.000	0.465 ± 0.000	$\mathbf{0.304 \pm 0.000}$
Pancreas	0.537 ± 0.001	0.506 ± 0.001	$\mathbf{0.352 \pm 0.000}$
Mouse Bladder	0.374 ± 0.000	0.374 ± 0.001	$\mathbf{0.170 \pm 0.000}$
Zeisel	0.580 ± 0.001	0.538 ± 0.000	$\mathbf{0.489 \pm 0.000}$
Worm Neuron	0.330 ± 0.000	0.190 ± 0.000	$\mathbf{0.049 \pm 0.000}$
Baron Human	0.493 ± 0.000	0.475 ± 0.000	$\mathbf{0.328 \pm 0.000}$

D Missing Not at Random Settings

Existing studies [11, 12] simulate dropout by randomly sampling non-zero values in a cell-gene matrix from a uniform distribution and setting them to zero (*i.e.*, missing completely at random (MCAR)). However, in real scRNA-seq data, dropouts occur more frequently in genes with low expression levels rather than those with high variance [33]. This is because the probability of capturing RNA transcripts of low-expression-level genes during sequencing is lower. Based on this dropout pattern, we selected the 1,000 genes with the lowest expression levels and simulated dropout only in these genes. We randomly sampled non-zero values of these genes from a uniform distribution and replaced the sampled values with zero (*i.e.*, missing not at random (MNAR)).

Table 4 presents the performance comparison under the aforementioned MNAR settings in terms of data recovery, measured by RMSE. We compared our scCR to the two most competitive baselines, scFP [11] and scBFP [12]. The dropout rate was set to 20% of the total number of values in the cell-gene matrix. As shown in the table, scCR outperformed the compared methods by significant margins in the realistic dropout settings, demonstrating the robustness of scCR in realistic scenarios. Considering realistic dropout simulation can help pre-assess the generalizability of techniques in practical scRNA-seq applications.

E Memory Usage Analysis

Table 5: Comparison of input and memory complexity. $\mathbf{X} \in \mathbb{R}^{G \times C}$ denote a cell-gene matrix, where C and G represent the number of cells and genes, respectively. $\mathbf{A}^{cell} \in \mathbb{R}^{C \times C}$ and $\mathbf{A}^{gene} \in \mathbb{R}^{G \times G}$ denote cell-cell and gene-gene adjacency matrices, respectively. θ denotes trainable parameters. B represents the batch size for batch-wise k-NN graph construction.

Method	Input	Big-O
scTAG	$\mathbf{X}$, $\mathbf{A}^{cell}$, θ	$O(GC) + O(\mathcal{E}_{cell}) + O(\theta)$
DCA	$\mathbf{X}$, θ	$O(GC) + O(\theta)$
AutoClass	$\mathbf{X}$, θ	$O(GC) + O(\theta)$
scGNN 2.0	$\mathbf{X}$, θ	$O(GC) + O(\mathcal{E}_{cell}) + O(\theta)$
scGCL	$\mathbf{X}$, $\mathbf{A}^{cell}$, θ	$O(GC) + O(\mathcal{E}_{cell}) + O(\theta)$
MAGIC	$\mathbf{X}$, $\mathbf{A}^{cell}$	$O(GC) + O(\mathcal{E}_{cell})$
scFP	$\mathbf{X}$, $\mathbf{A}^{cell}$	$O(BC) + O(\mathcal{E}_{cell})$
scBFP	$\mathbf{X}$, $\mathbf{A}^{cell}$, $\mathbf{A}^{gene}$	$O(BC) + O(\mathcal{E}_{gene}) + O(BG) + O(\mathcal{E}_{cell})$
scCR (Ours)	$\mathbf{X}$, $\mathbf{A}^{cell}$, $\mathbf{A}^{gene}$	$O(BC) + O(\mathcal{E}_{gene}) + O(BG) + O(\mathcal{E}_{cell})$

We analyzed the memory complexity of all methods used in this paper and conducted additional experiments to examine the memory usage of our scCR. Table 5 compares the input and memory complexity of scCR with other state-of-the-art methods. To alleviate the high memory demands during k-NN graph construction, we adopted the batch-wise k-NN graph construction strategy from [12]. When constructing k-NN graphs among genes, we divided the genes into batches of size B and computed k-nearest neighbors for each batch. We applied the same batch-wise strategy when constructing k-NN graphs among cells. This approach reduces memory requirements by avoiding the need to store distances between all points in the entire dataset simultaneously. Specifically, in the memory complexity of scCR, batch-wise k-NN graph construction changes $O(G^2)$ and $O(C^2)$ to $O(BG)$ and $O(BC)$, respectively. Consequently, batch-wise k-NN graph construction enables

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

	Baron Mouse	Pancreas	Mouse Bladder	Zeisel	Worm Neuron	Baron Human
Memory usage (GB)	1.811	1.837	1.957	2.037	1.927	3.861

processing of large datasets that would otherwise be infeasible due to memory constraints. Moreover, scCR does not require any trainable parameters, unlike other deep-learning-based methods.

We further measured the memory usage of scCR across various datasets, as shown in Table 6. The results in the table indicate that the advantages of scCR extend beyond its superior performance and time efficiency, showcasing its scalability as well.

F Hyperparameter Sensitivity

Table 7: Performance of scCR on cell clustering measured by ARI for different values of α.

α	Baron Mouse	Zeisel	Baron Human
0.01	0.627 ± 0.000	0.903 ± 0.000	0.823 ± 0.000
0.05 (used)	0.827 ± 0.093	0.902 ± 0.000	0.823 ± 0.000
0.1	0.727 ± 0.141	0.904 ± 0.000	0.824 ± 0.000
0.5	0.701 ± 0.000	0.901 ± 0.000	0.681 ± 0.000
0.9	0.448 ± 0.001	0.825 ± 0.000	0.683 ± 0.001

Table 8: Performance of scCR on cell clustering measured by ARI for different values of β.

β	Baron Mouse	Zeisel	Baron Human
0.1	0.440 ± 0.046	0.659 ± 0.058	0.553 ± 0.040
0.5	0.476 ± 0.049	0.724 ± 0.000	0.560 ± 0.001
0.9	0.509 ± 0.004	0.740 ± 0.000	0.657 ± 0.000
0.95	0.498 ± 0.002	0.910 ± 0.000	0.666 ± 0.011
0.99 (used)	0.827 ± 0.093	0.902 ± 0.000	0.823 ± 0.000
0.999	0.925 ± 0.000	0.900 ± 0.000	0.819 ± 0.000

Table 9: Performance of scCR on cell clustering measured by ARI for different values of γ.

γ	Baron Mouse	Zeisel	Baron Human
0.001	0.927 ± 0.001	0.902 ± 0.000	0.824 ± 0.000
0.01 (used)	0.827 ± 0.093	0.902 ± 0.000	0.823 ± 0.000
0.05	0.635 ± 0.000	0.903 ± 0.000	0.822 ± 0.000
0.1	0.595 ± 0.000	0.903 ± 0.000	0.667 ± 0.001
0.5	0.469 ± 0.003	0.740 ± 0.000	0.627 ± 0.011
0.9	0.409 ± 0.009	0.749 ± 0.000	0.586 ± 0.006

Table 10: Performance of scCR on cell clustering measured by ARI for different values of k.

k	Baron Mouse	Zeisel	Baron Human
1	0.628 ± 0.000	0.905 ± 0.000	0.827 ± 0.000
2 (used)	0.827 ± 0.093	0.902 ± 0.000	0.823 ± 0.000
3	0.631 ± 0.000	0.903 ± 0.000	0.819 ± 0.000
5	0.625 ± 0.002	0.902 ± 0.001	0.818 ± 0.000
10	0.621 ± 0.000	0.903 ± 0.000	0.818 ± 0.000
15	0.630 ± 0.000	0.902 ± 0.000	0.817 ± 0.000

We conducted additional experiments to provide a comprehensive analysis of the impact of different hyperparameters, including α, β, γ, and k, on the performance of scCR. We report ARI in cell clustering on three datasets by varying α, β, γ, and k within the ranges of {0.01, 0.05, 0.1, 0.5,

0.9}, {0.1, 0.5, 0.9, 0.95, 0.99, 0.999}, {0.001, 0.01, 0.05, 0.1, 0.5, 0.9}, and {1, 2, 3, 5, 10, 15}, respectively. When varying a target parameter, other hyperparameters were fixed to their default settings. Table 7, Table 8, Table 9, and Table 10 illustrate how these hyperparameter choices impact scCR performance. As shown in the tables, the values of α, β, γ, and k used in this study generally resulted in strong performance.

In terms of sensitivity, when the runner-up ARI scores are 0.660 ± 0.00, 0.848 ± 0.00, and 0.677 ± 0.00 for Baron Mouse, Zeisel, Baron Human, respectively, scCR demonstrated robustness against hyperparameter variations. Specifically, $\alpha \in \{0.05, 0.1, 0.5\}$, $\beta \in \{0.99, 0.999\}$, and $\gamma \in \{0.001, 0.01\}$ yielded state-of-the-art performance across the datasets. For k, scCR showed strong performance across all values, except in the case of Baron Human. Additionally, some parameter adjustments led to performance surpassing that of the default settings. However, given the unsupervised nature of single-cell analysis, we retain default hyperparameter settings that generally perform well.

G Experimental Details

G.1 Implementation Details

We conducted all the experiments on a single NVIDIA GeForce RTX 2080 Ti GPU and an Intel Core I5-6600 CPU at 3.30 Hz. The number of neighbors (k) in cell-cell and gene-gene k-NN graphs were set to 15 and 2, respectively. The total number of propagation steps K was set to 40 for both cell-cell and gene-gene FP. We set α, β, and γ to 0.05, 0.99, and 0.01, respectively. We found that the choice between row-stochastic normalization and symmetric normalization applied to $\overline{\mathbf{A}}^{cell(3)}$ within Soft FP [11] affected performance, and we reported the best result. For dropout recovery, we excluded the denoising stage (*i.e.*, $\gamma = 1$).

G.2 Datasets

For our experiments, we utilized six real scRNA-seq datasets, including Baron Mouse [23], Pancreas [24], Mouse Bladder [25], Zeisel [2], Worm Neuron [26], and Baron Human [23]. Table 11 summarizes the dataset statistics. We downloaded all the datasets from the GitHub repository[2] for [11]. These publicly available datasets and the repository have no public declaration of license.

We leveraged a commonly used pre-processing procedure for scRNA-seq data as described in recent studies [11, 12]. We performed minimal quality control (QC) using SCANPY [34], a toolkit for scRNA-seq analyses. Cells and genes exhibiting no gene expression (*i.e.*, with all zero values) were removed from a given cell-gene matrix. We retained the 2,000 genes with the highest variance in each dataset. We then normalized each cell by total counts over all genes to ensure that every cell had an equal total count of 1.0. That is, every row vector was divided by its library size, which is the sum of its values. After scaling by the median library size, $\log(x + 1)$ transformation was applied to all values in the cell-gene matrix, resulting in a pre-processed cell-gene matrix.

Table 11: Dataset statistics.

Dataset	Protocol	#Cells	#Genes	#Cell Type
Baron Mouse	inDrop	1,886	14,861	13
Pancreas	inDrop	1,937	15,575	14
Mouse Bladder	Microwell-seq	2,746	19,771	16
Zeisel	STRT-seq UMI	3,005	19,972	7
Worm Neuron	sci-RNA-seq	4,186	13,488	10
Baron Human	inDrop	8,569	17,499	14

[2] https://github.com/Junseok0207/scFP

G.3 Baselines

For all the baselines, we used the code released by the author of the respective papers. Table 12 shows the URL links for the baselines. scTAG and scGNN 2.0 are under the MIT license. The licenses of DCA, AutoClass, and MAGIC are Apache-2.0, GPL-3.0, and GPL-2.0, respectively. The code for scGCL, scFP, and scBFP has no public declaration of license. For each baseline, we adhered the hyperpameter/parameter setting in the released code or its respective paper.

Table 12: URL links for baselines.

Baseline	URL Link
scTAG	https://github.com/Philyzh8/scTAG
DCA	https://github.com/theislab/dca
AutoClass	https://github.com/datapplab/AutoClass
scGNN 2.0	https://github.com/OSU-BMBL/scGNN2.0
scGCL	https://github.com/zehaoxiong123/scGCL
MAGIC	https://github.com/KrishnaswamyLab/MAGIC
scFP	https://github.com/Junseok0207/scFP
scBFP	https://github.com/Junseok0207/scBFP

G.4 Evaluation Metrics

G.4.1 Clustering

Higher ARI, NMI, and CA indicate better performance in cell clustering.

ARI. The Adjusted Rand Index (ARI) is the corrected-for-chance version of the Rand Index (RI). RI is computed as follows:

$$RI = \frac{TP + TN}{\binom{N}{2}} \tag{12}$$

where TP is the number of true positives and TN is the number of true negatives. True positive indicates the number of cell pairs correctly assigned to the same cluster, and TP indicates the number of cell pairs correctly assigned to different clusters. ARI can be calculated as follows:

$$ARI = \frac{RI - \mathbb{E}(RI)}{\max(RI) - \mathbb{E}[RI]} \tag{13}$$

While RI produces a value between 0 and 1, ARI can produce negative values if the index is less than the expected index.

NMI. Normalized Mutual Information (NMI) is a normalization of the Mutual Information (MI) score to scale the scores between 0 and 1. NMI is computed as follows:

$$NMI = \frac{2 \times I(S;C)}{H(S) + H(C)} \tag{14}$$

where S is ground-truth cell types, $I(\cdot,\cdot)$ is the mutual information between two input distributions, and $H(\cdot)$ is the entropy function. Here, all logs are base-2. Higher NMI indicates the distribution of predicted cluster distribution is more similar to ground-truth cell type distribution.

CA. Clustering Accuracy is calculated as follows:

$$CA = \max_{m} \frac{\sum_{i=1}^{N} \mathbb{1}_{s_i = m(c_i)}}{N} \tag{15}$$

where s_i is the ground-truth cell type of the i-th cell, c_i is predicted cluster assignment of the i-th cell, and $m(\cdot)$ is the matching function responsible for mapping predicted cluster assignments to ground-truth cell types.

G.4.2 Recovery

Lower Median L1 Distance and RMSE indicate better performance in data recovery. Consider two set $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_n\}$, where $\mathcal{X}$ is the set of imputed values and $\mathcal{Y}$ is the set of their ground-truth values.

Median L1 Distance. Median L1 Distance is calculated as follows:

$$Median\ L1\ Distance = median(|x_1 - y_1|, \ldots, |x_n - y_n|). \tag{16}$$

RMSE. Root Mean Square Error (RMSE) is computed as follows:

$$RMSE(\mathcal{X}, \mathcal{Y}) = \sqrt{\frac{\sum_{i=1}^{N}(x_i - y_i)^2}{N}} \tag{17}$$

H Additional Experimental Results

H.1 Varying Scales across Genes

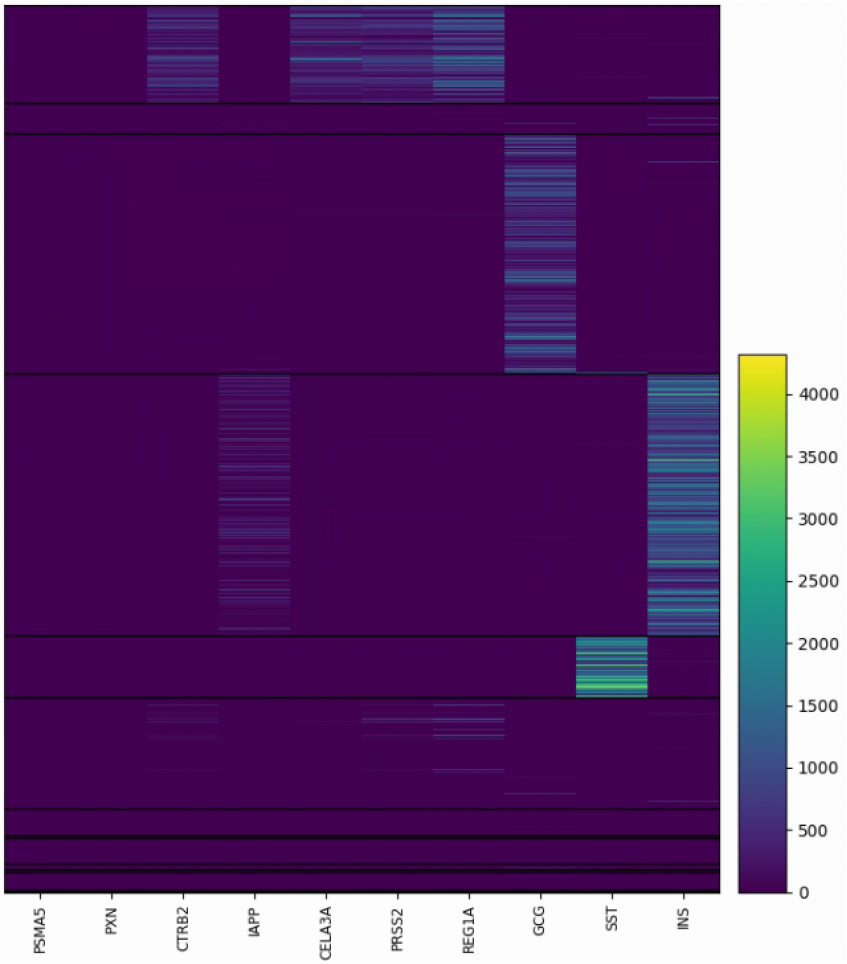

Figure 9: An heatmap of the cell-gene matrix in the Baron Human dataset. We randomly selected 10 genes (columns).

H.2 Dropout Recovery

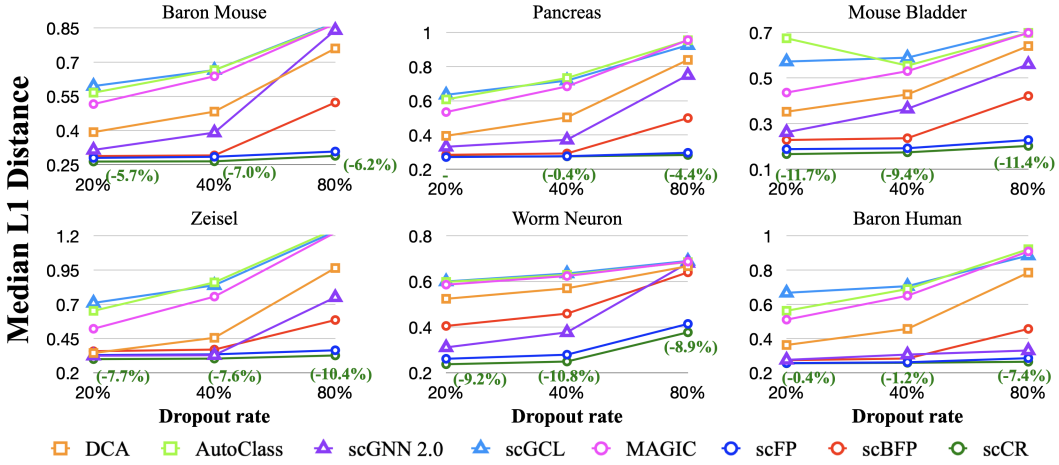

Figure 10: Performance on dropout recovery, measured by L1 Median Distance. Figures highlighted in green indicate reduction rates from the most competitive baseline at each setting.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: We clarified the scope of our work in the abstract and Sec. 1. The contributions of our paper are summarized in Sec. 1.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discussed the limitations of our work in Sec. 6 as follows: Like other scRNA-seq imputation methods, scCR is specifically designed for scRNA-seq data.

   Guidelines:

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

3. **Theory Assumptions and Proofs**

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

Answer: [Yes]

Justification: We provide the proof of convergence of FP in Appendix B.

Guidelines:

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

4. **Experimental Result Reproducibility**

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

   Answer: [Yes]

   Justification: We provide all the information needed to reproduce the main experimental results of our work. See Appendix G.1 and Appendix G.2.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We provide the publicly available sources of datasets and baselines in Appendix G.2 and Table 12 in Appendix G.3, respectively.

Guidelines:

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

6. **Experimental Setting/Details**

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

Answer: [Yes]

Justification: We specify all the training and test details in Sec. 5.1 and Appendix G.1.

Guidelines:

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

7. **Experiment Statistical Significance**

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

Answer: [Yes]

Justification: We provide the average performance with standard deviation errors across independent three runs. See Table 1 and Table 2.

Guidelines:

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

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

8. **Experiments Compute Resources**

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

   Answer: [Yes]

   Justification: We provide detailed information on the computer resources in Appendix G.1.

   Guidelines:

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

9. **Code Of Ethics**

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

   Answer: [Yes]

   Justification: Our research conforms with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader Impacts**

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

    Answer: [Yes]

    Justification: We discussed both potential positive societal impacts and negative societal impacts of our work in the Broader Impacts section.

    Guidelines:

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

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

11. **Safeguards**

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

    Answer: [NA]

    Justification: This paper poses no such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We specify the sources and licenses for all the baselines and datasets in Appendix G.3 and Appendix G.2, respectively.

    Guidelines:

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

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

13. **New Assets**

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

    Answer: [NA]

    Justification: This paper does not release new assets.

    Guidelines:

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

14. **Crowdsourcing and Research with Human Subjects**

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

    Answer: [NA]

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

    Guidelines:

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

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

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

    Answer: [Yes]

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

    Guidelines:

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

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