GeST: Towards Building A Generative Pretrained Transformer for Learning Cellular Spatial Context

Minsheng Hao[1]* **Nan Yan**[1]* **Haiyang Bian**[1]* **Yixin Chen**[1]
Jin Gu[1] **Lei Wei**[1] **Xuegong Zhang**[1]†
[1]Department of Automation, Tsinghua University
{hms20, yann21, bhy22, chenyx19}@mails.tsinghua.edu.cn,
{jgu, weilei92, zhangxg}@tsinghua.edu.cn

Abstract

Learning spatial context of cells through pretraining on spatial transcriptomics (ST) data may empower us to decipher tissue organization and cellular interactions. Yet, transformer-based generative models often focus on modeling individual cells, overlooking the intricate spatial relationships within them. To address this limitation, we develop GeST, a deep transformer model pretrained by a novel spatially informed generation task: Predicting cellular expression profile of a given location based on the information from its neighboring cells. GeST integrates a specialized spatial attention mechanism for efficient pretraining, a flexible serialization strategy for sequentializing ST data, and a cell tokenization method for quantizing gene expression profiles. We pretrained GeST on large-scale ST datasets across multiple ST technologies, achieving superior performance in generating previously unseen spatial cell profiles, extracting spatial niche embeddings in a zero-shot manner, and annotating spatial regions. Furthermore, GeST can simulate gene expression changes in response to perturbations of cells within spatial context, closely matching existing experimental results. GeST offers a powerful generative pre-training framework for learning spatial contexts.

1 Introduction

Transformer-based models pre-trained on large-scale data have emerged as a new paradigm in AI for biology [30, 6, 25], allowing the development of foundation models tailored to specific modalities such as DNA sequences [20], proteins [1] and single cell gene expression [26, 13, 8, 3]. However, most of these models focus on gene-gene relationships within the isolated cell, neglecting the intricate cell-cell communications. They struggle to handle spatial tasks such as understanding spatial cell patterns, which limits their ability to comprehend cellular behaviors in complex tissue systems.

Spatial transcriptomics (ST) offers high-throughput gene expression profiling with spatial localization of cells within tissue sections [19], which would be critical for deciphering tissue organization mechanisms [21, 32] and has great value in biological and medical research, such as identification of therapeutic biomarkers [35]. Previous studies, such as SpaGCN[15] and GraphST[17] applied graph neural networks to integrate gene expression and spatial information for learning cells' spatial representations. These models were trained independently for each dataset, leaving the paradigms of pretraining or generative modeling unexplored.

*Equal contribution

†Corresponding author

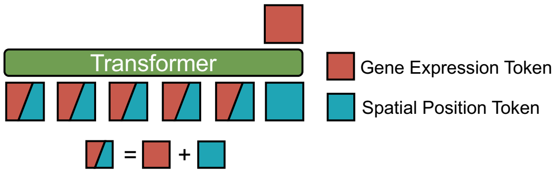

Figure 1: Given the spatial location x of one target cell, the transformer model takes its spatial coordinate $s(x)$, its neighbors' $N(x)$ gene expression $g(N(x))$ and spatial location $s(N(x))$ as the input, and predicts the target cell's gene expression $g(x)$.

Graph foundation models [34, 4] have recently emerged, leveraging masked modeling to capture spatial context. Transformer-based approaches have also been explored. For example, CellPLM [31] introduces a BERT-style [9] pre-trained model that uses partial gene expression data from a target cell, together with information from its neighboring cells, to predict the remaining gene expression. However, it cannot generate entirely new cells at unseen spatial locations, limiting its ability to investigate how spatial context shapes cellular characteristics. Moreover, BERT-style architecture predicts all outputs simultaneously from the given input, without the flexibility to adapt to dynamic spatial contexts, which further constrains downstream applications such as in silico spatial perturbation.

Inspired by the advancement of GPT models [2, 23, 5], we develop a generative pre-trained transformer and propose a spatially informed generation task for ST data, in which the model's objective is to iteratively generate cells' expression at given positions based on information of their neighboring cells (Figure 1). Building such a model faces several unique challenges. First, ST data are known for their irregular data structure, where cells are distributed arbitrarily in 2-dimensional tissue sections with varying numbers. Methods like Vision Transformer [11], which consider 2D data as a sequence with a fixed number and order, are not suitable. Second, directly generating continuous gene expression values of cells may introduce error accumulation during the autoregressive inference [22]. A robust tokenization method is needed to map cells into discrete tokens, analogous to the tokens used in the large language model.

To address these challenges, we present **GeST**, a deep **Ge**nerative pre-trained transformer for **ST** data which generates cells by leveraging the neighbor information. After pre-training on a large corpus of ST data, our model can learn biologically meaningful spatial patterns and can also be fine-tuned for other applications. Moreover, it can explore perturbation effects in spatial contexts by manipulating the given neighborhood information. To the best of our knowledge, GeST is the first generative pre-trained transformer to understand cell-cell relationships and advance cell modeling in the spatial context. Taken together, our work makes the following key contributions:

- **Spatially Informed Generation of ST Data**: We develop novel modeling for generative pre-trained transformers on ST data. To implement this task with high computational efficiency and flexibility, we design an attention mechanism called *Spatial Attention* and a serialization strategy to convert ST data into a sequence.
- **Cell Tokenization Method**: We develop a cell tokenization method to quantize cells' expression profiles to discrete tokens, along with a hierarchical pre-training loss designed to mitigate error accumulation in autoregressive generation.
- **Superior performance in Downstream Tasks**: We conducted several experiments and comparisons to show GeST superior performance in generating unseen spatial cells, spatial clustering and annotation tasks.
- **Spatial Perturbation Analysis**: We establish GeST as a pioneering model for *in-silico* spatial perturbation analysis. The simulation results align with real experiments, which provide an *in-silico* extension of current single-cell perturbation studies.

2 Task Formulation

Given a spatial transcriptomics dataset, we denote it as a set $\{x_1, x_2, x_3, \ldots, x_n\}$, encompassing all n cells within a 2-dimensional tissue slice. We define two essential functions: $g(\cdot)$, the gene expression retrieval function, and $s(\cdot)$, the spatial information retrieval function. We also define that for any given cell x, the set $N(x) = \{x_{N1}, x_{N2}, \ldots, x_{Nk}\}$ denotes all k of its neighboring cells.

Spatially informed generation (Pretraining task). We propose a generation pretraining task to learn cell-cell relationships within the spatial context. Given a target cell x, the objective is to predict the gene expression $g(x)$ based on its spatial location $s(x)$ and the information of its neighboring cells $N(x)$, i.e., conditional distribution:

$$P\left(g(x)|s(x), g(N(x)), s(N(x))\right) \tag{1}$$

Following this modeling, the objective function is

$$\min_{\theta} ||g(x) - \mathcal{F}_{\theta}(x \mid s(x), g(N(x)), s(N(x)))|| \tag{2}$$

where $\mathcal{F}_{\theta}$ represents our proposed spatial generative model parameterized by θ. However, spatial data lack a natural sequential order, which challenges the application of auto-regressive models that usually work for next token prediction tasks. To address this, we devise a serialization strategy to convert a set of cells $N(x_{k+1})$ into an ordered sequence $(x_1, x_2, \ldots, x_k)$. Then, we can transform the objective function into a sequential format:

$$\min_{\theta} ||g(x_{k+1}) - \mathcal{F}_{\theta}(x_{k+1} \mid s(x_{k+1}), g(x_1), s(x_1), g(x_2), s(x_2), \ldots, g(x_k), s(x_k))|| \tag{3}$$

Similar to natural language generation, models optimized for this task objective can apply the function $\mathcal{F}$ of Equation 3 iteratively to generate the gene expression of cells adjacent to the known cell neighbors.

Niche clustering/annotation (Downstream tasks). The concept 'niche' refers to a functional or structural tissue region where cells interact with each other and their surroundings. Identifying and understanding these niches is crucial for elucidating tissue organization [16]. The objective of niche clustering or annotation task is to extract the information of a cell x and its neighbors $N(x)$ into a latent embedding $E_{\phi}(x, N(x))$ that can be used for clustering or niche label classification:

$$E_{\phi}(x, N(x)) = \mathcal{F}_{\phi}(g(x), s(x), g(N(x)), s(N(x))) \tag{4}$$

3 Methodology

GeST has three components: an ST data serialization strategy, a cell tokenization method, and a spatial context-aware decoder. (Figure 2).

3.1 Spatial Context-Aware Decoder

Our model employs a decoder-only transformer architecture[27]. During pre-training, the model input comprises two contiguous token sequences: a neighbor cell sequence and a target cell position sequence, together totaling $2N - 2$ tokens. The model outputs a sequence predicting the gene expression for the $N - 1$ subsequent cells (Figure 2a).

Neighbor Cell Sequence. This sequence includes complete embeddings of the first $(N - 1)$ cells, integrating both gene expression and spatial position: $[gs(x_1), gs(x_2), \ldots, gs(x_{N-1})]$, where each token $gs(x_i) = g(x_i) + s(x_i)$ combines the gene expression $g(x_i)$ and spatial position embedding $s(x_i)$ for cell x_i.

Target Cell Position Sequence. This sequence contains only spatial position tokens for cells from the second to the N-th positions: $[s(x_2), s(x_3), \ldots, s(x_N)]$.

For each spatial token $s(x_{i+1})$ in the input sequence, the model predicts the gene expression $g(x_{i+1})$ of the corresponding target cell. This prediction is conditioned on previous complete neighbor tokens $gs(x_1), gs(x_2), \ldots, gs(x_i)$ and the current spatial position token $s(x_{i+1})$.
To efficiently capture spatial dependencies, we introduce **Spatial Attention**, a specialized attention mechanism. Given a sequence length $2L$ (where $L = N - 1$), we define an attention mask M as a $2L \times 2L$ matrix. For the token at position $i + L$, corresponding to predicting the gene expression of cell x_{i+1}, attention is

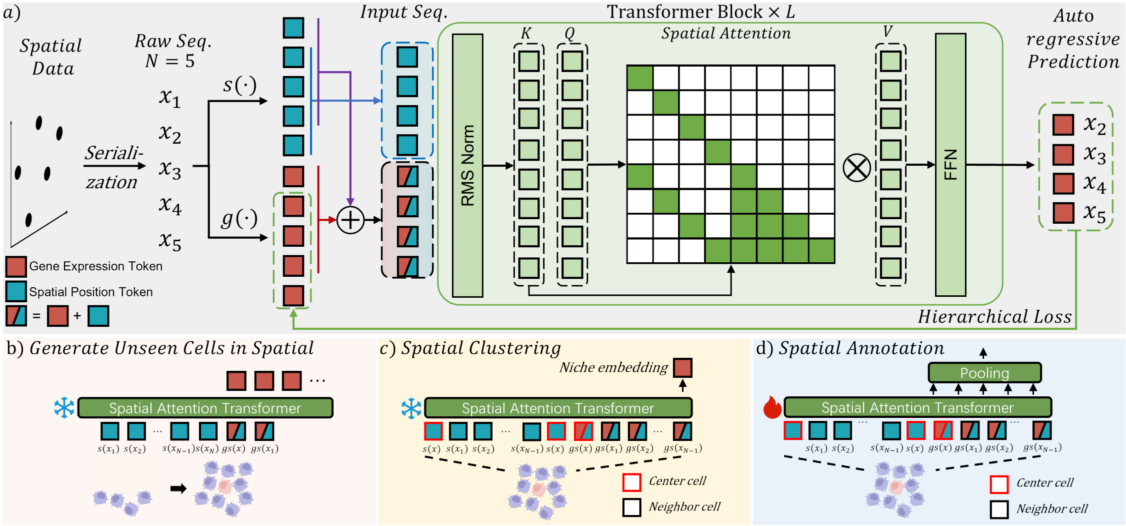

Figure 2: Schematic overview of GeST. a) Model architecture. b) Generating unseen cells in spatial from neighborhood cells c) Extracting a niche embedding from a group of neighborhood cells and doing spatial clustering d) Annotating spatial niches by fine-tuning GeST.

restricted to: 1) Neighbor cell tokens at positions 1 to i (i.e., $gs(x_1), gs(x_2), \ldots, gs(x_i)$), and 2) The spatial position token at position $i + L$ (i.e., $s(x_{i+1})$).

Formally, for $i \in [1, L]$, the attention mask M is defined as:

$$M_{i+L,t} = \begin{cases} 1, & \text{if } t \in \{i, L+1, L+2, \ldots, L+i\} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

As Spatial Attention limits token interactions to a suffix sequence, it enables parallel model training, effectively capturing spatial context relationships[23]. After the decoder, a multilayer perceptron is used to convert the hidden embedding $\mathbf{h} \in \mathbb{R}^D$ to gene expression space $\hat{\mathbf{y}} \in \mathbb{R}^T$. Each element of prediction $\hat{\mathbf{y}}$ represents the expression level of a specific gene, where T is the total number of genes.

3.2 Serialization strategy

To generate the input sequence at each training step, we crop a square region from the training tissue section and define a set of all N cells within this square as $\mathcal{X} = x_{o1}, x_{o2}, \ldots, x_{oN}$. Cells are serialized by sampling along diagonal paths. Specifically, a cell from the square's four vertices is randomly chosen as the starting point x_1. We then calculate the Euclidean distances between x_1 and all other cells as sampling weights: $\{w_{o1}, w_{o2}, w_{o3}, ..., w_{oN}\}$, where $w_{oi} = ||p(x_{oi}) - p(x_1)||_2$ and $p(x) = (p_1, p_2)$ denotes the 2D coordinates of the cell. We do sampling without replacement for $N - 1$ times to get a sequence $(x_1, x_2, x_3, ...x_N)$. At each sampling time t, the probability of selecting cell x_{oi} is:

$$P(x_t = x_{oi}) = \frac{w_{oi}}{\sum_{j \in \mathcal{X} \setminus \mathcal{D}} w_{oj}} \tag{6}$$

where $\mathcal{D}$ contains cells that have been selected into the sequence. This strategy ensures spatially adjacent cells have similar indices, maintaining order randomness to prevent overfitting.

3.3 Cell tokenization

Functions $g(x), s(x)$ tokenize a cell's gene expression and spatial position into d dimensions. For the spatial position $s(x)$, we set up a coordinate system whose origin is the center cell of the tissue section. Then we calculate the other cells' relative coordinates. We tokenize the coordinate values by a 2D sinusoidal positional encoding (SPE) function SPE: $\mathbb{R}^2 \to \mathbb{R}^d$ similar to ViT [11].

For gene expressions $g(x)$, directly using continuous vectors as targets led to error accumulation and model instability in our preliminary experiments (see "w/o quantization" in Table 4). Thus, we propose a "meta cell vocabulary" $\mathcal{C}$ to quantize continuous expressions into discrete tokens. Given a training dataset of n cells and T genes, we apply PCA to reduce dimensions to p, cluster cells into K groups using K-means, and designate each cluster center as a "meta cell". Each meta cell c_i possesses a mean PCA vector $\mathcal{C}\text{pca}[i]$ and a mean gene expression vector $\mathcal{C}\text{expr}[i]$.

To tokenize cell x, we identify its nearest meta cell in PCA space and assign the corresponding meta cell's gene expression as its token:

$$g(x) = \mathcal{C}\text{expr}[i], \quad i = \arg\min_{k} \|x\text{pca} - \mathcal{C}_{\text{pca}}[k]\|_2 \tag{7}$$

3.4 Loss function

To compute the loss between the predicted gene expression vector $\hat{\mathbf{y}}$ and the ground truth gene expression token $g(x)$, we propose a hierarchical cross-entropy loss function. We perform K-means clustering on the meta cell vocabulary using a smaller number of clusters to obtain hierarchical labels.

Specifically, each meta cell $c \in \mathcal{C}$ is assigned hierarchical labels at four levels: $l_0(c)$, $l_1(c)$, $l_2(c)$, and $l_3(c)$, with corresponding numbers of categories K, K_1, K_2, and K_3. The level $l_0(c)$ corresponds to the original fine-grained meta cell vocabulary. We project model outputs $\hat{\mathbf{y}}$ into logits $\mathbf{z} \in \mathbb{R}^K$ using: $\mathbf{z} = \hat{\mathbf{y}}\mathcal{C}_{expr}^{\top}$, where $\mathcal{C}_{expr} \in \mathbb{R}^{K \times T}$ holds the expression profiles of meta cells. At level i, the probability for label k is computed as:

$$p^{(i)}(k) = \sum_{c \in \mathcal{C}} \delta\left(l_i(c) = k\right) p(c), \quad p(c) = \frac{\exp(z_c)}{\sum_{c' \in \mathcal{C}} \exp(z_{c'})} \tag{8}$$

where $\delta(\cdot)$ is the Kronecker delta function, which equals 1 if the condition is true and 0 otherwise. $p(c)$ is the predicted probability that $\hat{\mathbf{y}}$ corresponds to meta cell c. The overall loss function $\mathcal{L}$ is defined as a weighted sum of the negative log-likelihood losses at each hierarchical level:

$$\mathcal{L} = \sum_{i=0}^{3} \alpha_i \cdot \mathcal{L}_i = \sum_{i=0}^{3} \alpha_i \cdot \left(-l_i(x) \log p^{(i)}\right) \tag{9}$$

where α_i are weights and we set 0.25 as default. $l_i(x)$ is the ground truth label of the target cell's nearest meta cell at level i. By minimizing this hierarchical loss function, the model is encouraged to make correct predictions at multiple levels, making it robust to single level wrong predictions.

Regarding inference (Figure 2b), there are two modes to translate predicted probabilities into a final gene expression vector: (1) "Picking" mode: Select the expression profile of the meta cell with the highest predicted probability. (2) "Weighted aggregation" mode: Aggregate expression profiles from all meta cells, weighted by their predicted probabilities $p(c)$.

3.5 Niche embedding extraction

After pretraining, GeST enables niche clustering in a zero-shot manner and can be fine-tuned for niche annotation. Both tasks require the extraction of a niche embedding. Given a niche with a target cell x and its $N-1$ neighbors, we first input position tokens and cell tokens. The target cell's position token $s(x)$ appears twice—once at the beginning and once at the end—creating $N+1$ position tokens. We incorporate the content tokens for all N cells (Figure 2c). The model's final output token, encapsulating information from all cells, defines the niche embedding (Equation 4), used for zero-shot niche clustering.

For fine-tuning, we input the sequence in the same setting and further apply a mean pooling operation to aggregate these embeddings into a single vector (Figure 2d) $\mathbf{h}_p = \text{Pool}(\{\mathbf{h}_x\} \cup \{\mathbf{h}_n \mid n \in N(x)\})$.

Table 1: Benchmark on unseen cell generation. R.all: Root mean square error (RMSE) of all genes. R.200: Root mean square error of top 200 SVGs. ρ: Spearman's rank correlation coefficient. The best scores of each testing set are in **bold**, and the second best are in underline.

Method	**MERFISH**									**Visium**			**Stereo-seq**		
	Anterior Brain			Mid Brain			Posterior Brain			Primary Liver Cancer			Sagittal Brain		
	R.all↓	R.200↓	$\rho\uparrow$	R.all↓	R.200↓	$\rho\uparrow$	R.all↓	R.200↓	$\rho\uparrow$	R.all↓	R.200↓	$\rho\uparrow$	R.all↓	R.200↓	$\rho\uparrow$
Ours	1.374	1.296	0.301	1.379	1.319	0.265	1.386	1.340	0.242	**1.303**	**1.126**	**0.540**	1.405	1.357	**0.326**
Ours-W	**1.352**	**1.244**	**0.340**	**1.367**	**1.288**	**0.302**	**1.376**	**1.322**	**0.275**	1.320	1.150	0.499	**1.399**	**1.340**	0.323
GP	1.379	1.289	0.241	1.389	1.337	0.241	1.393	1.354	0.214	1.357	1.264	0.272	1.413	1.405	0.073
MLP	1.399	1.369	0.319	1.404	1.393	0.283	1.406	1.397	0.267	1.347	1.174	0.491	1.403	1.357	0.314

4 Experiments

We conducted four experiments for demonstration: unseen cell generation, zero-shot niche embedding clustering, fine-tuning-based niche label annotation, and *in-silico* spatial perturbation. Sec4.1 shows our model's superior performance in the spatial generation task. Sec4.2 and Sec4.3 show the generalization ability to other downstream applications. Sec4.4 explores GeST's ability in an *in-silico* perturbation study.

4.1 Unseen cell generation

To validate the generative capability of our model, we used three datasets from different ST technologies and resolutions covering normal and disease samples: a single-cell resolution MERFISH dataset of the whole mouse brain [36], a multi-cell resolution Visium dataset of human primary liver cancer (PLC) [32], and a sub-cell resolution Stereo-seq dataset of the mouse brain sagittal section [7]. For each dataset, we divided all tissue sections into training, validation, and testing sets (see supplementary for detailed descriptions). We cropped a region of the section in the testing set and named it as an "unseen region". Our task is to predict the gene expression of this unseen region based on the remaining part of the section. Since there are no existing methods designed for spatial generation of ST data, we trained two models as baselines: a gaussian process (GP) model and a multilayer perceptron (MLP). These two models used cells' absolute spatial coordinates and gene expressions from the uncropped areas in each section as training data.

Recognizing that not all genes exhibit strong spatial patterns, we focused on the top 200 spatially variable genes (SVGs) per slide, identified using SOMDE [14]. As shown in Table 1), our model in the "picking" mode (labeled as "Ours") achieved lower root mean square error (RMSE) of both all genes and top 200 SVGs compared to the baseline models. Switched to the "weighted aggregation" mode (labeled as "Ours-W"), our model produced even lower regression error in the MERFISH dataset and consistently surpassed baselines in the Spearman's coefficient. The supplementary materials contain more visualization results. These findings highlight our model's capability of learning the underlying spatial characteristics of gene expression and cell organization through the spatially informed generation task.

4.2 Unsupervised niche clustering

We evaluated GeST capability on spatial niche clustering and annotation tasks in the MERFISH mouse brain dataset, as it is well-annotated with two levels of anatomical labels, "Division" and "Region", according to the Mouse Brain Common Coordinate Framework v3 [29]. We compared our method with four methods: NicheCompass, GraphST, STAGATE and SpaGCN. We also included a baseline that gets cluster labels based solely on the cell's own gene expression without spatial information (Raw). As shown in the Table 2, all spatial clustering methods except SpaGCN outperformed the raw baseline on the adjusted mutual information (AMI) scores. Our pretrained model (labeled as "Ours") achieved higher AMI than other graph-based methods in most test sets. As expected, after we continued training our model on the test data in the same

Table 2: Benchmark of AMI scores on unsupervised niche clustering tasks. NicheC: NicheCompass. The best scores of each testing set are in **bold**, and the second best are in underline.

Method	Anterior Brain		Mid Brain		Posterior Brain	
	Division	Region	Division	Region	Division	Region
Ours	0.344	**0.406**	0.550	0.535	0.460	0.472
Ours-FT	**0.368**	0.404	**0.585**	**0.589**	**0.489**	**0.528**
GraphST	0.220	0.361	0.485	0.463	0.405	0.365
NicheC	0.253	0.386	0.547	0.551	0.449	0.440
STAGATE	0.229	0.382	0.527	0.516	0.447	0.449
SpaGCN	0.126	0.191	0.244	0.265	0.207	0.196
Raw	0.083	0.183	0.233	0.288	0.216	0.219

generative way (labeled as “Ours-FT"), the model showed consistently higher performance. These results demonstrate that our pre-trained model can be effectively transferred to new tissues in a zero-shot manner.

4.3 Supervised Niche annotation

Currently, spatial annotation methods are lacking due to limited data with fine annotation. Therefore, we compared two single cell annotation methods: scANVI [33] and Celltypist [10], which only utilize gene expression of one cell for classification. As detailed in Table 3, our model greatly outperformed the single-cell methods at both resolutions. This is because the niche labels were annotated in terms of both gene expression and spatial location information of every cell, and our model perceives all the cells in their spatial context rather than treating them individually. A showcase annotation result is in the supplementary figure 6.7.

Table 3: Benchmark of F1 score on niche annotation task. The best scores of each testing set are in **bold**.

Method	Anterior Brain		Mid Brain		Posterior Brain	
	Division	Region	Division	Region	Division	Region
Ours	**0.711**	**0.483**	**0.549**	**0.392**	**0.462**	**0.312**
scANVI	0.309	0.222	0.337	0.216	0.264	0.126
CellTypist	0.106	0.032	0.205	0.066	0.131	0.042

4.4 In-silico spatial perturbation

Measuring the cell response by perturbations like diseases or treatments is critical in biological and medical research [24]. Here, we establish GeST as a pioneer model for predicting cell response of *in-silico* perturbation in spatial. Compared with other generative models[28, 18] in spatial, GeST can simulate the spatial effects that arise when perturbations are applied to groups of cells, rather than only to individual cells.

We conducted the experiment based on a mouse brain ischemic study (Figure 3a). In the original study, Han *et al.* [12] identified the infarct core area (ICA) and the proximal region of the peri-infarct area (PIA_P) from the data. We chose an area with a similar location to ICA from our MERFISH dataset

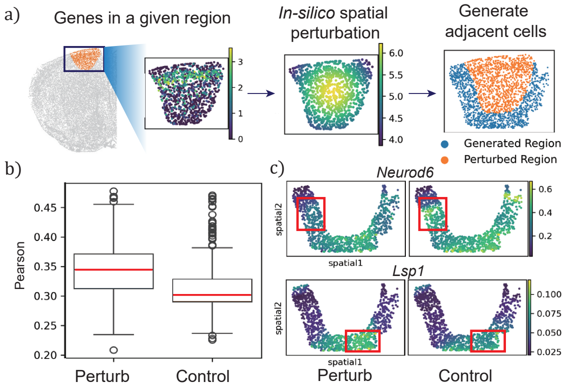

Figure 3: a) Overview of *in-silico* spatial perturbation experiment. b) PCCs of the perturbation and control group. P-value < 0.001, t-test. c) Visualization of two predicted DEGs.

as a given region, and manually altered gene expression in this given region according to the differentially expressed genes (DEGs) identified in ICA. Then we fed the altered data to our model to generate the perturbed gene expression of adjacent cells of the given region, representing the predicted PIA_P. To validate the results, we calculated the Pearson correlation coefficient (PCC) between the predicted and real PIA_P gene expression, and compared it with the control group. The perturbation group showed significantly higher PCCs (Figure 3b). Taking all 87 high and low DEGs in PIA_P as ground truth, we correctly classified 70.11% of them by our *in-silico* perturbation experiment. This is higher than the baseline accuracy of 44.8%, which is obtained by simply adopting the DEGs from ICA (i.e., a naive model that believes changes in ROI are the same in the neighbor). Figure 3c gives two predicted DEG examples. These findings are consistent with the experimental results from [12].

4.5 Ablation study

We conducted comprehensive ablation experiments to assess the effects of model size, training data volume, neighbor window size, and designed modules on performance (Table 4). Increasing the model size from a smaller size to our default setup resulted in substantial improvements, whereas further enlargement yielded diminishing gains. This result suggests that our baseline model provides an optimal balance between computational cost and predictive accuracy. Additionally, models trained on larger datasets outperformed those with smaller datasets. These results indicate the scalability of our model architecture.

The neighbor window size also influenced model outcomes by controlling input information density and sequence length. Models with a window size of 800μm outperformed 200μm window size, yet increasing the window size alone did not yield further improvement. However, increasing both window and model size (800μm+L12H8) achieved the best results, suggesting that larger spatial contexts introduce complexity requiring greater model capacity. All these results reveal that our default setting represents a balanced compromise between window size and model complexity.

Table 4: Ablation Results. 'Ours' is the default model with 8 layers and 8 heads (L8H8) trained on full data and 600μm window size. '800μm+L' is a model with L12H8 trained on 800μm window size. 'w/o spatial info' stands for replacing all positional embedding with an all-ones vector. R.all: RMSE of all genes. R.50: RMSE of top 50 SVGs. ρ: Spearman's rank correlation coefficient.

	Ours	**Model Size**			**Data**		**Window Size**			**w/o hierarchy**	**w/o quantization**	**w/o spatial info**	**random order**
		L2H2	L4H4	L16H16	1/2	1/3	200μm	800μm	800μm+L				
R.all↓	1.367	1.371	1.373	1.362	1.381	1.375	1.376	1.371	1.362	1.382	1.389	1.397	1.384
R.50↓	1.214	1.243	1.249	1.208	1.261	1.246	1.251	1.232	1.204	1.270	1.315	1.325	1.256
$\rho\uparrow$	0.29	0.288	0.289	0.291	0.289	0.288	0.275	0.285	0.292	0.288	N.A.	0.230	0.287

Removing hierarchical loss decreased performance across all metrics. Replacing the quantization module with a mean squared error (MSE) loss resulted in invalid negative predictions and poor RMSE, making Spearman correlation calculation impossible. Omitting spatial neighbor information also caused significant performance degradation. Finally, we compared our spatial ordinal serialization strategy with random sampling. Our ordinal strategy showed considerable performance gains, supporting our hypothesis that serialization aligning with actual spatial patterns enhances prediction accuracy.

5 Conclusion

We present GeST, a deep generative transformer model pretrained by a novel spatially informed generation task. GeST features a spatial attention mechanism, paired with a serialization strategy and cell tokenization module, enabling several downstream tasks. To the best of our knowledge, GeST is the first generative pretrained model in spatial transcriptomics. We believe GeST would lay the groundwork for building comprehensive foundation models for spatial biology.

References

[1] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pages 1–3, 2024.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[3] Haiyang Bian, Yixin Chen, Xiaomin Dong, Chen Li, Minsheng Hao, Sijie Chen, Jinyi Hu, Maosong Sun, Lei Wei, and Xuegong Zhang. scmulan: a multitask generative pre-trained language model for single-cell analysis. In *International Conference on Research in Computational Molecular Biology*, pages 479–482. Springer, 2024.

[4] Quentin Blampey, Hakim Benkirane, Nadege Bercovici, Fabrice André, and Paul-Henry Cournede. Novae: a graph-based foundation model for spatial transcriptomics data. *bioRxiv*, pages 2024–09, 2024.

[5] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[6] Charlotte Bunne, Yusuf Roohani, Yanay Rosen, Ankit Gupta, Xikun Zhang, Marcel Roed, Theo Alexandrov, Mohammed AlQuraishi, Patricia Brennan, Daniel B Burkhardt, et al. How to build the virtual cell with artificial intelligence: Priorities and opportunities. *arXiv preprint arXiv:2409.11654*, 2024.

[7] Mengnan Cheng, Liang Wu, Lei Han, Xin Huang, Yiwei Lai, Jiangshan Xu, Shuai Wang, Mei Li, Huiwen Zheng, Weimin Feng, et al. A cellular resolution spatial transcriptomic landscape of the medial structures in postnatal mouse brain. *Frontiers in Cell and Developmental Biology*, 10:878346, 2022.

[8] Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature Methods*, pages 1–11, 2024.

[9] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[10] C Domínguez Conde, C Xu, LB Jarvis, DB Rainbow, SB Wells, T Gomes, SK Howlett, O Suchanek, K Polanski, HW King, et al. Cross-tissue immune cell analysis reveals tissue-specific features in humans. *Science*, 376(6594):eabl5197, 2022.

[11] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[12] Bing Han, Shunheng Zhou, Yuan Zhang, Sina Chen, Wen Xi, Chenchen Liu, Xu Zhou, Mengqin Yuan, Xiaoyu Yu, Lu Li, et al. Integrating spatial and single-cell transcriptomics to characterize the molecular and cellular architecture of the ischemic mouse brain. *Science Translational Medicine*, 16(733):eadg1323, 2024.

[13] Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Xuegong Zhang, and Le Song. Large-scale foundation model on single-cell transcriptomics. *Nature Methods*, pages 1–11, 2024.

[14] Minsheng Hao, Kui Hua, and Xuegong Zhang. Somde: a scalable method for identifying spatially variable genes with self-organizing map. *Bioinformatics*, 37(23):4392–4398, 2021.

[15] Jian Hu, Xiangjie Li, Kyle Coleman, Amelia Schroeder, Nan Ma, David J Irwin, Edward B Lee, Russell T Shinohara, and Mingyao Li. Spagcn: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nature methods*, 18(11):1342–1351, 2021.

[16] Sanjay Jain and Michael T Eadon. Spatial transcriptomics in health and disease. *Nature Reviews Nephrology*, pages 1–13, 2024.

[17] Yahui Long, Kok Siong Ang, Mengwei Li, Kian Long Kelvin Chong, Raman Sethi, Chengwei Zhong, Hang Xu, Zhiwei Ong, Karishma Sachaphibulkij, Ao Chen, et al. Spatially informed clustering,

integration, and deconvolution of spatial transcriptomics with graphst. *Nature Communications*, 14(1):1155, 2023.

[18] Stathis Megas, Daniel G Chen, Krzysztof Polanski, Moshe Eliasof, Carola-Bibiane Schonlieb, and Sarah A Teichmann. Celcomen: spatial causal disentanglement for single-cell and tissue perturbation modeling. *arXiv preprint arXiv:2409.05804*, 2024.

[19] Lambda Moses and Lior Pachter. Museum of spatial transcriptomics. *Nature methods*, 19(5):534–546, 2022.

[20] Eric Nguyen, Michael Poli, Matthew G Durrant, Armin W Thomas, Brian Kang, Jeremy Sullivan, Madelena Y Ng, Ashley Lewis, Aman Patel, Aaron Lou, et al. Sequence modeling and design from molecular to genome scale with evo. *BioRxiv*, pages 2024–02, 2024.

[21] Giovanni Palla, David S Fischer, Aviv Regev, and Fabian J Theis. Spatial components of molecular tissue biology. *Nature Biotechnology*, 40(3):308–318, 2022.

[22] Marco Pasini, Javier Nistal, Stefan Lattner, and George Fazekas. Continuous autoregressive models with noise augmentation avoid error accumulation. *arXiv preprint arXiv:2411.18447*, 2024.

[23] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[24] Jennifer E Rood, Anna Hupalowska, and Aviv Regev. Toward a foundation model of causal cell and tissue biology with a perturbation cell and tissue atlas. *Cell*, 187(17):4520–4545, 2024.

[25] Artur Szałata, Karin Hrovatin, Sören Becker, Alejandro Tejada-Lapuerta, Haotian Cui, Bo Wang, and Fabian J Theis. Transformers in single-cell omics: a review and new perspectives. *Nature Methods*, 21(8):1430–1443, 2024.

[26] Christina V Theodoris, Ling Xiao, Anant Chopra, Mark D Chaffin, Zeina R Al Sayed, Matthew C Hill, Helene Mantineo, Elizabeth M Brydon, Zexian Zeng, X Shirley Liu, et al. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

[27] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[28] Chloe Wang, Haotian Cui, Andrew Zhang, Ronald Xie, Hani Goodarzi, and Bo Wang. scgpt-spatial: Continual pretraining of single-cell foundation model for spatial transcriptomics. *bioRxiv*, pages 2025–02, 2025.

[29] Quanxin Wang, Song-Lin Ding, Yang Li, Josh Royall, David Feng, Phil Lesnar, Nile Graddis, Maitham Naeemi, Benjamin Facer, Anh Ho, et al. The allen mouse brain common coordinate framework: a 3d reference atlas. *Cell*, 181(4):936–953, 2020.

[30] Sarah Webb et al. Deep learning for biology. *Nature*, 554(7693):555–557, 2018.

[31] Hongzhi Wen, Wenzhuo Tang, Xinnan Dai, Jiayuan Ding, Wei Jin, Yuying Xie, and Jiliang Tang. Cellplm: pre-training of cell language model beyond single cells. *bioRxiv*, pages 2023–10, 2023.

[32] Rui Wu, Wenbo Guo, Xinyao Qiu, Shicheng Wang, Chengjun Sui, Qiuyu Lian, Jianmin Wu, Yiran Shan, Zhao Yang, Shuai Yang, et al. Comprehensive analysis of spatial architecture in primary liver cancer. *Science Advances*, 7(51):eabg3750, 2021.

[33] Chenling Xu, Romain Lopez, Edouard Mehlman, Jeffrey Regier, Michael I Jordan, and Nir Yosef. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular systems biology*, 17(1):e9620, 2021.

[34] Yuning You, Zitong Jerry Wang, Kevin Fleisher, Rex Liu, and Matt Thomson. Building foundation models to characterize cellular interactions via geometric self-supervised learning on spatial genomics. *bioRxiv*, pages 2025–01, 2025.

[35] Linlin Zhang, Dongsheng Chen, Dongli Song, Xiaoxia Liu, Yanan Zhang, Xun Xu, and Xiangdong Wang. Clinical and translational values of spatial transcriptomics. *Signal Transduction and Targeted Therapy*, 7(1):111, 2022.

[36] Meng Zhang, Xingjie Pan, Won Jung, Aaron R Halpern, Stephen W Eichhorn, Zhiyun Lei, Limor Cohen, Kimberly A Smith, Bosiljka Tasic, Zizhen Yao, et al. Molecularly defined and spatially resolved

cell atlas of the whole mouse brain. *Nature*, 624(7991):343–354, 2023.