Exploring Augmentation-Driven Invariances for Graph Self-supervised Learning in Spatial Omics

Lovro Rabuzin* **Michel Tarnow*** **Valentina Boeva**
Department of Computer Science, ETH Zurich
{lrabuzin, mtarnow, vboeva}@ethz.ch

Abstract

Spatial omics technologies provide rich insights into biological processes by jointly capturing molecular profiles and the spatial organization of cells. The resulting high-dimensional data can be naturally represented as graphs, where Graph Neural Networks (GNNs) offer an effective framework to model interactions in the tissue. Self-supervised pretraining methods leverage graph augmentations to build invariances without costly labels. Yet, the design of augmentation strategies remains underexplored, particularly in the context of spatial omics. In this work, we investigate how different graph augmentations affect embedding quality and downstream performance in spatial omics. We evaluate a suite of existing and novel augmentations, including transformations tailored to biological variation, across two representative tasks: unsupervised domain identification in healthy tissue and supervised phenotype prediction in cancer tissue. Our results show that carefully chosen augmentations substantially improve performance, whereas poorly aligned or overly complex augmentations may fail to help or even degrade performance.

1 Introduction

Spatial omics technologies measure molecular profiles, such as RNA or protein expression, while preserving the spatial context of cells in their natural environment. This modality provides a more comprehensive view of biological processes compared to non-spatial single-cell methods [1]. Spatial transcriptomics platforms use microscopy or in situ sequencing to generate spatial maps of RNA expression. Complementary proteomics methods measure protein abundances with spatial resolution.

The complex and high-dimensional data produced by these technologies can be naturally represented as graphs, where nodes correspond to cells and edges encode spatial proximity or molecular similarity [2]. To exploit all available information from spatial omics data, graph-based methods like graph neural networks (GNNs) often exhibit superior characteristics compared to traditional analysis methods not taking spatial dependencies in the data into account [3, 4]. GNNs are well-suited to analyze spatial omics data, as they explicitly model relationships between cells through graph structures using a message-passing mechanism [3].

Pretraining enables GNNs to learn generalizable patterns from data before fine-tuning them for specific tasks. Moreover, these approaches can introduce inductive biases, for instance via graph augmentations, that help models prioritize biologically relevant features and improve robustness [5].

A central principle of contrastive self-supervision is that it enforces invariance to augmentations: two different views of the same input are trained to have similar embeddings [6–8]. Recent work has shown that in graph domains, augmentations explicitly inject desired invariances, such as robustness to node/edge perturbations or feature corruption [5, 9].

*Equal contribution. Code available at https://github.com/BoevaLab/spatial-augmentations

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Several pretraining frameworks have operationalized these ideas in the graph setting. Deep Graph Contrastive Representation Learning (GRACE) [9] builds invariance to structural and feature perturbations via an InfoNCE-based contrastive loss. Bootstrapped Graph Latents (BGRL) [10] achieves similar invariances without negative samples, relying instead on online–target encoder consistency. While recent benchmarks such as scSSL-Bench [11] have evaluated self-supervised learning in a biological context across diverse single-cell omics modalities, spatial omics remains underexplored. Most existing applications of GNNs to spatial omics adopt generic augmentations from other domains or do not leverage augmentation at all [2, 12, 13]. However, spatial omics graphs differ fundamentally from social or molecular networks: their nodes represent spatially embedded cells, and edges encode physical proximity rather than arbitrary connectivity. As a result, conventional augmentations like random edge or feature drops can disrupt biologically meaningful tissue structure and spatial organization [14]. Effective pretraining in this domain therefore requires augmentation strategies that preserve spatial coherence while accounting for biological and experimental variability.

We explore how different graph augmentation strategies affect the performance of GNNs in spatial omics data. We investigate existing graph augmentations and newly designed augmentations that encode biologically meaningful inductive biases. Their effectiveness is evaluated on two downstream tasks in spatial omics: unsupervised domain identification on healthy mouse brain tissue and supervised phenotype prediction in human lung cancer samples. These tasks differ not only in supervision regime but also in biological complexity: domain identification on healthy tissue emphasizes stable spatial compartments, while phenotype prediction on cancer tissue must contend with tissue heterogeneity and noisy clinical labels [15–17]. We aim to quantify how different augmentations influence downstream performance. To our knowledge, this is the first systematic investigation of graph augmentations in spatial omics.

2 Methods

Our study design consists of applying graph augmentations (baseline and advanced) to input graphs, pretraining models using BGRL and GRACE, and evaluating on two downstream tasks: domain identification and phenotype prediction.

Notations A graph is denoted by $\mathbf{G} = (\mathbf{X}, \mathbf{A})$, where $\mathbf{X} \in \mathbb{R}^{N \times F}$ is the node feature matrix with N nodes and F features per node, and $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the binary adjacency matrix. Graph augmentations generate a new view $\tilde{\mathbf{G}} = (\tilde{\mathbf{X}}, \tilde{\mathbf{A}})$ by modifying $\mathbf{X}$, $\mathbf{A}$, or both. Node positions are encoded in a spatial matrix $\mathbf{P} \in \mathbb{R}^{N \times d}$, with $d = 2$, yielding $\tilde{\mathbf{P}}$ after augmentation. The neighborhood of node i, denoted $\mathcal{N}(i)$, is defined as the set of nodes directly connected to i in $\mathbf{A}$.

2.1 Baseline augmentations

Two baseline augmentations were used: **DropFeatures** and **DropEdges**. Models trained with these augmentations served as baselines for performance comparisons.

DropFeatures randomly masks features by setting entries in $\mathbf{X}$ to zero with probability p, resulting in $\tilde{\mathbf{X}}$ while keeping $\mathbf{A}$ unchanged. If $\mathbf{X}$ contains a cell type feature, it is masked out.

DropEdges randomly removes edges from $\mathbf{A}$ with probability p (Bernoulli sampling), resulting in $\tilde{\mathbf{A}}$ while keeping $\mathbf{X}$ unchanged.

2.2 Advanced augmentations

Advanced augmentations include both published and novel methods. These were tested individually and in combination to assess their effect on downstream tasks relative to baseline augmentations.

DropImportance drops features/edges with probabilities p derived from normalized log-degree importance I. Inspired by prior work [18, 19], it is controlled by dropout rate μ and threshold λ_p. Let

$$I_i^{(n)} = \frac{\log(1 + \deg_i) - \bar{d}}{\max_j \log(1 + \deg_j) - \bar{d}}, \quad p_i = \min\{(1 - I_i^{(n)})\,\mu,\ \lambda_p\}$$

where $\deg_i$ is the degree of node i and $\bar{d}$ is the mean log-degree. For edges, we use endpoint-mean importance and analogous dropout:

$$I_{ij}^{(e)} = \tfrac{1}{2}\big(I_i^{(n)} + I_j^{(n)}\big), \quad p_{ij} = \min\{(1 - I_{ij}^{(e)})\,\mu,\ \lambda_p\}.$$

This enforces invariance to removing less-informative node features and edges.

SpatialNoise adds Gaussian noise to spatial positions:

$$\tilde{\mathbf{p}}_i = \mathbf{p}_i + \boldsymbol{\epsilon}_i, \quad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}). \tag{1}$$

This models experimental imprecision in cell localization and enforces invariance to small spatial perturbations. This augmentation is applicable only to domain identification.

FeatureNoise adds Gaussian noise to node features:

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\epsilon}_i, \quad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}). \tag{2}$$

This simulates variability in molecular readouts and enforces robustness to fluctuations in expression.

SmoothFeatures applies a convex combination of each node's features with the mean of its neighbors:

$$\tilde{\mathbf{x}}_i = (1 - \alpha)\mathbf{x}_i + \alpha \cdot \tfrac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \mathbf{x}_j, \tag{3}$$

where $\alpha \in [0, 1]$ controls the smoothing strength. This simulates transcript leakage [20] and enforces invariance to local feature diffusion. This augmentation is used only for domain identification.

PhenotypeShift randomly mutates discrete cell-type features c_i according to a transition map $\mathcal{M}$:

$$\tilde{c}_i = \begin{cases} c_i & \text{with probability } 1 - p, \\ \text{sample}(\mathcal{M}[c_i]) & \text{with probability } p, \end{cases} \tag{4}$$

where $\mathcal{M}[c_i] \subseteq \mathcal{C}$ contains plausible phenotype alternatives. This models both plasticity (cell-type switching) and misclassification noise, training robustness to annotation uncertainty. This augmentation is used only for phenotype prediction. Details of $\mathcal{M}$ are dataset-specific.

2.3 The task of domain identification

The first task employed to evaluate augmentations is unsupervised *Domain Identification*. The objective is to detect and segment spatially coherent regions within healthy tissue based on molecular data (e.g., gene expression) and spatial data (e.g., spatial relationships). These regions, or domains, ideally reflect biologically relevant structures such as tissue compartments or functional zones.

We used three spatial transcriptomics datasets with expert domain annotations, obtained via the benchmarking study of Schaub *et al.* (2025) [21]. Dataset 1 profiles 5 mouse brain samples via MERFISH [22]. Dataset 2 contains STARmap data from mouse cortex [23], with expert annotations by Li and Zhou (2022) [24]. Dataset 3 comprises BaristaSeq samples of mouse cortex tissue [25]. All datasets are publicly available [26].

Data preprocessing and graph construction Each sample is first preprocessed using a sequence of filtering and normalization steps standard for spatial omics data [27]. Principal Component Analysis (PCA) is subsequently applied to the processed expression matrix. Following preprocessing, one spatial omics graph is constructed per sample. Each cell is represented as a node, with the top 50 principal components of gene expression serving as node features. Nodes are connected to their k nearest neighbors in Euclidean space. Parameter k is optimized during hyperparameter search.

Model and training A two-layer Graph Convolutional Network (GCN) is used to compute node embeddings for each sample-specific graph. The network is trained in a self-supervised manner using both the BGRL and GRACE frameworks. To encourage spatial coherence in the learned representations, a spatial regularization term is added. It penalizes high similarity in the embedding space for nodes that are spatially distant. This discourages long-range spurious similarities. The resulting overall loss function is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{SSL}} + \gamma_{\text{spatial}} \cdot \tfrac{1}{N^2} \sum_{i,j} \mathbf{D}_{i,j}^{(s)} \cdot (1 - \mathbf{D}_{i,j}^{(z)}), \tag{5}$$

where $\mathbf{D}^{(s)}_{i,j}$ denotes the normalized Euclidean distance between cells i and j, and $\mathbf{D}^{(z)}_{i,j}$ denotes the normalized distance between their embeddings in the latent space. The regularization strength is controlled by the hyperparameter γ_{spatial}.

Training is conducted across all data samples. For model selection and evaluation, the dataset is split into 40% validation and 60% test samples. All hyperparameters, including augmentation hyperparameters, were tuned on validation sets over fixed ranges (see Section A.8 in the Appendix).

Clustering To obtain the final domain assignments for each node, the learned node embeddings are clustered using the Leiden algorithm [28]. The resolution of the Leiden clustering is dynamically adjusted to match the number of ground truth domains in each sample. The predicted domain labels are evaluated against the ground truth annotations using clustering quality metrics (see Section A.5). Metrics are calculated per sample and averaged across the test set to report the performance. Reported means and standard deviations are computed over 5 independent runs with different random seeds.

2.4 The task of phenotype prediction

The second task used to evaluate augmentations is supervised *Phenotype Prediction* in human non-small cell lung cancer (NSCLC) tissue. The objective is to predict biological or clinical phenotypes directly from spatially resolved molecular data. Here, we predict cancer relapse after treatment.

The data used for phenotype prediction consists of one non-small cell lung cancer (NSCLC) spatial proteomics dataset obtained by imaging mass cytometry [29]. Marker expression was quantified in 1071 patients with at least 15 years follow-up, resulting in 1868 cancer samples. Each sample includes clinical annotations, for instance cancer stage, relapse, clinical outcome, or cancer subtype. The raw data can be downloaded from the resource provided by Cords *et al.* (2024) [29].

Data preprocessing and graph creation Graphs were constructed from segmented cells using Delaunay triangulation. Node features included cell type (integer-encoded) and cell size. Edge features consist of a binary "near/distant" category based on centroid-to-centroid distance, using a threshold of 20 μm, reflecting the typical size of human cells. From each tissue graph, h-hop subgraphs were extracted ($h = 3$ by default).

Model and training The phenotype prediction model is based on SPACE-GM [2]. An L-layer Graph Isomorphism Network (GIN) with edge-feature extension [2] was used, with messages

$$m^{(\ell)}_{vu} = h^{(\ell-1)}_u + e^{(\ell)}_{vu}, \tag{6}$$

with $e^{(\ell)}_{vu}$ mapped via an embedding lookup. Subgraph embeddings were obtained by max-pooling over final-layer node embeddings. The encoder was pretrained with BGRL and GRACE. For classification, a 3-layer MLP was added and jointly fine-tuned with a weighted BCE loss.

Splits and optimization Pretraining used all samples without labels. For supervised fine-tuning, 1492 samples were used for training and 376 for evaluation, with evaluation split into 50% validation and 50% test. The performance was evaluated against the ground truth patient labels using standard classification quality metrics (see Section A.4 in the Appendix). Hyperparameters were tuned on the validation set. Reported means and standard deviations are computed on the test set over 5 independent runs with different random seeds.

3 Results

3.1 Unsupervised domain identification in healthy mouse brain tissue

We evaluated the effect of augmentations on the task of identifying distinct domains in healthy mouse brain tissue. Baseline models were trained with *DropFeatures* and *DropEdges*, and compared against models with advanced augmentations or their combinations. The *Noise* augmentation denotes the joint application of *SpatialNoise* and *FeatureNoise*.

Results with BGRL and GRACE are shown in Tables 1 and 2. Under BGRL, *DropImportance* improved NMI from 0.61 (baseline) to 0.66, with the next-best performance achieved by combining

all augmentations (0.65). Under GRACE, *DropImportance* again achieved the best result (0.66 compared to 0.65 baseline), and was the only augmentation regime that substantially improved over the baseline. Across both frameworks, *DropImportance* provided the most consistent gains. With BGRL, nearly all augmentation regimes improved upon the baseline, whereas in GRACE the gains were smaller because the baseline was already comparatively strong.

Table 1: **Performance on domain identification task using BGRL.** Clustering performance on healthy mouse brain tissue using different augmentation strategies. Reported as mean ± standard deviation across 5 random seeds. The best and second-best results by mean are highlighted.

Augmentations	NMI	HOM	COM
Baseline	0.6145 ± 0.0195	0.6188 ± 0.0234	0.6121 ± 0.0175
Baseline + Noise	0.6488 ± 0.0083	0.6419 ± 0.0093	0.6576 ± 0.0074
DropImportance	**0.6585 ± 0.0033**	**0.6552 ± 0.0065**	**0.6635 ± 0.0008**
DropImportance + Noise	0.6488 ± 0.0166	0.6507 ± 0.0135	0.6498 ± 0.0217
SmoothFeatures	0.6497 ± 0.0065	0.6465 ± 0.0097	0.6538 ± 0.0061
DropImp. + Noise + SmoothFeat.	0.6540 ± 0.0104	0.6507 ± 0.0110	0.6579 ± 0.0103

Table 2: **Performance on domain identification task using GRACE.** Clustering performance on healthy mouse brain tissue using different augmentation strategies. Reported as mean ± standard deviation across 5 random seeds. The best and second-best results by mean are highlighted.

Augmentations	NMI	HOM	COM
Baseline	0.6470 ± 0.0081	0.6475 ± 0.0081	0.6484 ± 0.0110
Baseline + Noise	0.6405 ± 0.0221	0.6390 ± 0.0157	0.6438 ± 0.0271
DropImportance	**0.6639 ± 0.0056**	**0.6569 ± 0.0082**	**0.6726 ± 0.0046**
DropImportance + Noise	0.6477 ± 0.0125	0.6409 ± 0.0120	0.6557 ± 0.0127
SmoothFeatures	0.6460 ± 0.0100	0.6423 ± 0.0115	0.6509 ± 0.0085
DropImp. + Noise + SmoothFeat.	0.6412 ± 0.0058	0.6336 ± 0.0050	0.6502 ± 0.0080

Qualitative results of the models trained using BGRL are shown in Figure 1 for a representative MERFISH sample. Different augmentation strategies produce visibly different domain segmentations, broadly consistent with the quantitative metrics.

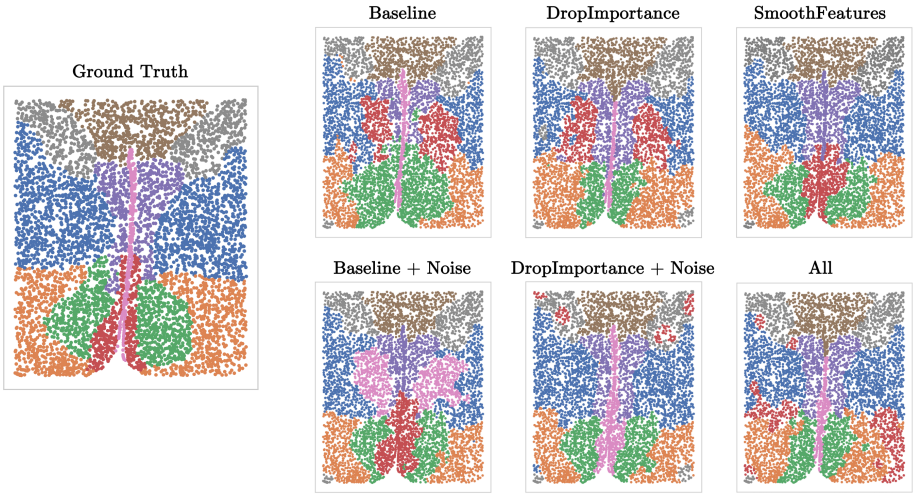

Figure 1: **Predicted and ground-truth domains in MERFISH tissue.** Visualization of a representative mouse brain sample. The left-most panel shows expert-annotated ground truth; remaining panels display predicted domains under different augmentation strategies. Augmentations strongly influence segmentation quality, broadly consistent with the quantitative results.

3.2 Supervised phenotype prediction in human cancer tissue

We evaluated augmentation strategies on relapse prediction in NSCLC samples, with performance measured by F1 score and AUROC (Tables 3 and 4). Models were trained with baseline augmentations (*DropFeatures* and *DropEdges*) and compared against advanced augmentations. In *PhenotypeShift*, we incorporated biologically motivated cell state transitions (see Section A.6 in the Appendix).

Under BGRL, the baseline achieved F1 = 0.59 and AUROC = 0.60. *FeatureNoise* improved AUROC to 0.61, while *DropImportance* alone decreased performance. The best F1 was obtained with *PhenotypeShift* (0.64), while the best AUROC was achieved by *FeatureNoise* (0.61). Combining all augmentations yielded intermediate gains (F1 = 0.62, AUROC = 0.60).

Under GRACE, the results showed similar patterns. The best F1 score was obtained with *DropImportance* + *FeatureNoise* (0.63), while the best AUROC was achieved with *Baseline* + *FeatureNoise* (0.59). The second best scores were achieved using *PhenotypeShift* for both metrics (F1 score 0.63, AUROC 0.59). Adding all augmentations together did not improve over single strategies.

Table 3: **Performance on phenotype prediction task using BGRL.** Relapse prediction in NSCLC samples using BGRL pretraining with different augmentation strategies. Reported as mean ± standard deviation across 5 random seeds. The best and second-best results by mean are highlighted.

Augmentations	F1 Score	AUROC
Baseline	0.5896 ± 0.0213	0.5986 ± 0.0142
Baseline + FeatureNoise	0.6265 ± 0.0155	**0.6084 ± 0.0097**
DropImportance	0.6171 ± 0.0245	0.5848 ± 0.0031
DropImportance + FeatureNoise	0.6277 ± 0.0011	0.5665 ± 0.0106
PhenotypeShift	**0.6375 ± 0.0090**	0.6006 ± 0.0098
DropImp. + FeatNoise + PhenotypeShift	0.6218 ± 0.0291	0.6030 ± 0.0100

Table 4: **Performance on phenotype prediction task using GRACE.** Relapse prediction in NSCLC samples using GRACE pretraining with different augmentation strategies. Reported as mean ± standard deviation across 5 random seeds. The best and second-best results by mean are highlighted.

Augmentations	F1 Score	AUROC
Baseline	0.6157 ± 0.0125	0.5759 ± 0.0194
Baseline + FeatureNoise	0.6208 ± 0.0142	**0.5932 ± 0.0090**
DropImportance	0.6137 ± 0.0355	0.5707 ± 0.0074
DropImportance + FeatureNoise	**0.6338 ± 0.0052**	0.5545 ± 0.0122
PhenotypeShift	0.6318 ± 0.0096	0.5897 ± 0.0167
DropImp. + FeatNoise + PhenotypeShift	0.6070 ± 0.0409	0.5870 ± 0.0088

4 Discussion

We systematically evaluated the role of graph augmentations in self-supervised GNN pretraining for spatial omics, using both BGRL [10] and GRACE [9] across two tasks: domain identification and phenotype prediction. While the absolute performance scores are modest, this primarily reflects the intrinsic difficulty and noise of these tasks in spatial omics data. Our models nonetheless reach performance levels comparable to recent state-of-the-art approaches, demonstrating the validity of the setup. The results show that augmentation choice has a decisive impact on downstream performance. In line with prior contrastive learning work [6–8], we find that well-aligned augmentations can enhance representations by encoding task-relevant invariances, whereas overly strong or misaligned transformations can degrade performance.

Domain identification benefited most from structural perturbations, with *DropImportance* improving performance by removing structurally redundant nodes and edges, albeit at the cost of additional computational overhead due to the need to compute and rank node and edge importance scores. In contrast, phenotype prediction showed limited gains from structural perturbations and instead improved with noise-based and biologically motivated augmentations such as *FeatureNoise* and

PhenotypeShift. Composing these augmentations provided only limited additional benefit or even hurt performance. A likely explanation is that the combined perturbations either dilute informative signal or exceed the capacity of the model to leverage additional invariances in this noisy, small-sample setting.

This study was limited to two downstream tasks and a curated set of augmentations. Moreover, the two tasks differed both in supervision regime and biological complexity, making it difficult to disentangle whether augmentation effectiveness depends primarily on task type (unsupervised vs. supervised) or tissue context (healthy vs. cancerous). Future work could address this by including supervised tasks on healthy tissue and unsupervised tasks on cancer tissue.

In summary, augmentation design is a critical factor in self-supervised learning on spatial omics graphs. Effective augmentations encode biologically plausible invariances, improving model robustness and downstream accuracy, while misaligned ones can add cost without benefit. Our results reinforce the view that augmentation choice is not incidental but a central design decision in graph contrastive learning.

References

[1] Dario Bressan, Giorgia Battistoni, and Gregory J. Hannon. The dawn of spatial omics. *Science*, 381(6657):4964, 2023. doi: 10.1126/science.abq4964. URL https://www.science.org/doi/abs/10.1126/science.abq4964.

[2] Zhenqin Wu, Alexandro E. Trevino, Eric Wu, Kyle Swanson, Honesty J. Kim, H. Blaize D'Angio, Ryan Preska, Gregory W. Charville, Piero D. Dalerba, Ann Marie Egloff, Ravindra Uppaluri, Umamaheswar Duvvuri, Aaron T. Mayer, and James Zou. Graph deep learning for the characterization of tumour microenvironments from spatial protein profiles in tissue specimens. *Nature Biomedical Engineering*, 6(12):1435–1448, November 2022. ISSN 2157-846X. doi: 10.1038/s41551-022-00951-w. URL http://dx.doi.org/10.1038/s41551-022-00951-w.

[3] Bharti Khemani, Shruti Patil, Ketan Kotecha, and Sudeep Tanwar. A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions. *Journal of Big Data*, 11(1):18, 2024. doi: 10.1186/s40537-023-00876-4. URL https://journalofbigdata.springeropen.com/articles/10.1186/s40537-023-00876-4.

[4] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural Message Passing for Quantum Chemistry. *arXiv e-prints*, art. arXiv:1704.01212, April 2017. doi: 10.48550/arXiv.1704.01212.

[5] Yue You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *arXiv preprint arXiv:2010.13902*, 2020. URL https://arxiv.org/abs/2010.13902.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. *arXiv e-prints*, art. arXiv:2002.05709, February 2020. doi: 10.48550/arXiv.2002.05709.

[7] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised Learning. *arXiv e-prints*, art. arXiv:2006.07733, June 2020. doi: 10.48550/arXiv.2006.07733.

[8] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What Makes for Good Views for Contrastive Learning? *arXiv e-prints*, art. arXiv:2005.10243, May 2020. doi: 10.48550/arXiv.2005.10243.

[9] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep Graph Contrastive Representation Learning. *arXiv e-prints*, art. arXiv:2006.04131, June 2020. doi: 10.48550/arXiv.2006.04131.

[10] Shantanu Thakoor, Corentin Tallec, Mohammad Gheshlaghi Azar, Rémi Munos, Petar Veličković, and Michal Valko. Large-scale representation learning on graphs via bootstrapping. *arXiv preprint arXiv:2102.06514*, 2021. URL https://arxiv.org/abs/2102.06514.

[11] Olga Ovcharenko, Florian Barkmann, Philip Toma, Imant Daunhawer, Julia Vogt, Sebastian Schelter, and Valentina Boeva. scSSL-Bench: Benchmarking Self-Supervised Learning for Single-Cell Data. *arXiv e-prints*, art. arXiv:2506.10031, June 2025. doi: 10.48550/arXiv.2506.10031.

[12] Yuxuan Hu, Jiazhen Rong, Yafei Xu, Runzhi Xie, Jacqueline Peng, Lin Gao, and Kai Tan. Unsupervised and supervised discovery of tissue cellular neighborhoods from cell phenotypes. *Nature Methods*, 21(2):267–278, January 2024. ISSN 1548-7105. doi: 10.1038/s41592-023-02124-2. URL http://dx.doi.org/10.1038/s41592-023-02124-2.

[13] Shay Shimonov, Joseph M Cunningham, Ronen Talmon, Lilach Aizenbud, Shruti J Desai, David Rimm, Kurt Schalper, Harriet Kluger, and Yuval Kluger. Sorbet: Automated cell-neighborhood analysis of spatial transcriptomics or proteomics for interpretable sample classification via gnn. January 2024. doi: 10.1101/2023.12.30.573739. URL http://dx.doi.org/10.1101/2023.12.30.573739.

[14] Kaize Ding, Zhe Xu, Hanghang Tong, and Huan Liu. Data Augmentation for Deep Graph Learning: A Survey. *arXiv e-prints*, art. arXiv:2202.08235, February 2022. doi: 10.48550/arXiv.2202.08235.

[15] Cyril Neftel, Julie Laffy, Mariella G. Filbin, Toshiro Hara, Marni E. Shore, Gilbert J. Rahme, Alyssa R. Richman, Dana Silverbush, McKenzie L. Shaw, Christine M. Hebert, John Dewitt, Simon Gritsch, Elizabeth M. Perez, L. Nicolas Gonzalez Castro, Xiaoyang Lan, Nicholas Druck, Christopher Rodman, Danielle Dionne, Alexander Kaplan, Mia S. Bertalan, Julia Small, Kristine Pelton, Sarah Becker, Dennis Bonal, Quang-De Nguyen, Rachel L. Servis, Jeremy M. Fung, Ravindra Mylvaganam, Lisa Mayr, Johannes Gojo, Christine Haberler, Rene Geyeregger, Thomas Czech, Irene Slavc, Brian V. Nahed, William T. Curry, Bob S. Carter, Hiroaki Wakimoto, Priscilla K. Brastianos, Tracy T. Batchelor, Anat Stemmer-Rachamimov, Maria Martinez-Lage, Matthew P. Frosch, Ivan Stamenkovic, Nicolo Riggi, Esther Rheinbay, Michelle Monje, Orit Rozenblatt-Rosen, Daniel P. Cahill, Anoop P. Patel, Tony Hunter, Inder M. Verma, Keith L. Ligon, David N. Louis, Aviv Regev, Bradley E. Bernstein, Itay Tirosh, and Mario L. Suvà. An integrative model of cellular states, plasticity, and genetics for glioblastoma. *Cell*, 178 (4):835–849.e21, August 2019. ISSN 0092-8674. doi: 10.1016/j.cell.2019.06.024. URL http://dx.doi.org/10.1016/j.cell.2019.06.024.

[16] Karin Pelka, Matan Hofree, Jonathan H. Chen, Siranush Sarkizova, Joshua D. Pirl, Vjola Jorgji, Alborz Bejnood, Danielle Dionne, William H. Ge, Katherine H. Xu, Sherry X. Chao, Daniel R. Zollinger, David J. Lieb, Jason W. Reeves, Christopher A. Fuhrman, Margaret L. Hoang, Toni Delorey, Lan T. Nguyen, Julia Waldman, Max Klapholz, Isaac Wakiro, Ofir Cohen, Julian Albers, Christopher S. Smillie, Michael S. Cuoco, Jingyi Wu, Mei-ju Su, Jason Yeung, Brinda Vijaykumar, Angela M. Magnuson, Natasha Asinovski, Tabea Moll, Max N. Goder-Reiser, Anise S. Applebaum, Lauren K. Brais, Laura K. DelloStritto, Sarah L. Denning, Susannah T. Phillips, Emma K. Hill, Julia K. Meehan, Dennie T. Frederick, Tatyana Sharova, Abhay Kanodia, Ellen Z. Todres, Judit Jané-Valbuena, Moshe Biton, Benjamin Izar, Conner D. Lambden, Thomas E. Clancy, Ronald Bleday, Nelya Melnitchouk, Jennifer Irani, Hiroko Kunitake, David L. Berger, Amitabh Srivastava, Jason L. Hornick, Shuji Ogino, Asaf Rotem, Sébastien Vigneau, Bruce E. Johnson, Ryan B. Corcoran, Arlene H. Sharpe, Vijay K. Kuchroo, Kimmie Ng, Marios Giannakis, Linda T. Nieman, Genevieve M. Boland, Andrew J. Aguirre, Ana C. Anderson, Orit Rozenblatt-Rosen, Aviv Regev, and Nir Hacohen. Spatially organized multicellular immune hubs in human colorectal cancer. *Cell*, 184(18):4734–4752.e20, September 2021. ISSN 0092-8674. doi: 10.1016/j.cell.2021.08.003. URL http://dx.doi.org/10.1016/j.cell.2021.08.003.

[17] Andrew L. Ji, Adam J. Rubin, Kim Thrane, Sizun Jiang, David L. Reynolds, Robin M. Meyers, Margaret G. Guo, Benson M. George, Annelie Mollbrink, Joseph Bergenstråhle, Ludvig Larsson, Yunhao Bai, Bokai Zhu, Aparna Bhaduri, Jordan M. Meyers, Xavier Rovira-Clavé, S. Tyler Hollmig, Sumaira Z. Aasi, Garry P. Nolan, Joakim Lundeberg, and Paul A. Khavari. Multimodal analysis of composition and spatial architecture in human squamous cell carcinoma. *Cell*, 182(2):497–514.e22, July 2020. ISSN 0092-8674. doi: 10.1016/j.cell.2020.05.039. URL http://dx.doi.org/10.1016/j.cell.2020.05.039.

[18] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Graph Contrastive Learning with Adaptive Augmentation. *CoRR*, 2010.14945, 2020. URL https://arxiv.org/abs/2010.14945.

[19] Yichun Li, Jin Huang, Weihao Yu, and Tinghua Zhang. Neighborhood-enhanced contrast for pre-training graph neural networks. *Neural Computing and Applications*, 36(8):4195–4205, 2024. doi: 10.1007/s00521-023-09274-6. URL https://link.springer.com/article/10.1007/s00521-023-09274-6.

[20] Yue You, Yuting Fu, Lanxiang Li, Zhongmin Zhang, Shikai Jia, Shihong Lu, Wenle Ren, Yifang Liu, Yang Xu, Xiaojing Liu, Fuqing Jiang, Guangdun Peng, Abhishek Sampath Kumar, Matthew E. Ritchie, Xiaodong Liu, and Luyi Tian. Systematic comparison of sequencing-based spatial transcriptomic methods. *Nature Methods*, 21(9):1743–1754, July 2024. ISSN 1548-7105. doi: 10.1038/s41592-024-02325-3. URL http://dx.doi.org/10.1038/s41592-024-02325-3.

[21] Darius P. Schaub, Behnam Yousefi, Nico Kaiser, Robin Khatri, Victor G. Puelles, Christian F. Krebs, Ulf Panzer, and Stefan Bonn. PCA-based spatial domain identification with state-of-the-art performance. *Bioinformatics*, 41(1):5, 2025. doi: 10.1093/bioinformatics/btaf005. URL https://academic.oup.com/bioinformatics/article/41/1/btaf005/7945104.

[22] Jeffrey R. Moffitt, Devjanee Bambah-Mukku, Stephen W. Eichhorn, Eric Vaughn, Karthik Shekhar, Jonathan D. Perez, Nimrod D. Rubinstein, Junjie Hao, Aviv Regev, Catherine Dulac, and Xiaowei Zhuang. Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. *Science*, 362(6416):eaat5324, 2018. doi: 10.1126/science.aau5324. URL https://www.science.org/doi/abs/10.1126/science.aau5324.

[23] Xiao Wang, William E. Allen, Matthew A. Wright, Emily L. Sylwestrak, Nikolay Samusik, Sam Vesuna, Kathryn Evans, Cindy Liu, Charu Ramakrishnan, Jia Liu, Garry P. Nolan, Felice-Alessio Bava, and Karl Deisseroth. Three-dimensional intact-tissue sequencing of single-cell transcriptional states. *Science*, 361(6400):5691, 2018. doi: 10.1126/science.aat5691. URL https://www.science.org/doi/abs/10.1126/science.aat5691.

[24] Zheng Li and Xiang Zhou. BASS: multi-scale and multi-sample analysis enables accurate cell type clustering and spatial domain detection in spatial transcriptomic studies. *Genome Biology*, 23(1):168, 2022. doi: 10.1186/s13059-022-02734-7. URL https://genomebiology.biomedcentral.com/articles/10.1186/s13059-022-02734-7.

[25] Brian Long, Jeremy Miller, and The SpaceTx Consortium. SpaceTx: A Roadmap for Benchmarking Spatial Transcriptomics Exploration of the Brain, 2023. URL https://arxiv.org/abs/2301.08436.

[26] Zhiyuan Yuan, Fangyuan Zhao, Senlin Lin, Yu Zhao, Jianhua Yao, Yan Cui, Xiao-Yong Zhang, and Yi Zhao. Benchmarking spatial clustering methods with spatially resolved transcriptomics data. *Nature Methods*, 21(4):712–722, 2024. doi: 10.1038/s41592-024-02215-8. URL https://www.nature.com/articles/s41592-024-02215-8.

[27] Lukas Heumos, Anna C. Schaar, Christopher Lance, Anastasia Litinetskaya, Felix Drost, Luke Zappia, Malte D. Lücken, Daniel C. Strobl, Juan Henao, Fabiola Curion, Hananeh Aliee, Meshal Ansari, Pau Badia-i Mompel, Maren Büttner, Emma Dann, Daniel Dimitrov, Leander Dony, Amit Frishberg, Dongze He, Soroor Hediyeh-zadeh, Leon Hetzel, Ignacio L. Ibarra, Matthew G. Jones, Mohammad Lotfollahi, Laura D. Martens, Christian L. Müller, Mor Nitzan, Johannes Ostner, Giovanni Palla, Rob Patro, Zoe Piran, Ciro Ramírez-Suástegui, Julio Saez-Rodriguez, Hirak Sarkar, Benjamin Schubert, Lisa Sikkema, Avi Srivastava, Jovan Tanevski, Isaac Virshup, Philipp Weiler, Herbert B. Schiller, and Fabian J. Theis. Best practices for single-cell analysis across modalities. *Nature Reviews Genetics*, 24(8):550–572, March 2023. ISSN 1471-0064. doi: 10.1038/s41576-023-00586-w. URL http://dx.doi.org/10.1038/s41576-023-00586-w.

[28] V. A. Traag, L. Waltman, and N. J. van Eck. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, 2019. doi: 10.1038/s41598-019-41695-z. URL https://www.nature.com/articles/s41598-019-41695-z.

[29] Lena Cords, Stefanie Engler, Martina Haberecker, Jan Hendrik Rüschoff, Holger Moch, Natalie de Souza, and Bernd Bodenmiller. Cancer-associated fibroblast phenotypes are associated with patient outcome in non-small cell lung cancer. *Cancer Cell*, 42(3):396–412, 2024. doi: 10.1016/j.ccell.2023.12.021. URL https://www.cell.com/cancer-cell/fulltext/S1535-6108(23)00449-X.

[30] Zhou Chen, Fangfang Han, Yan Du, Huaqing Shi, and Wence Zhou. Hypoxic microenvironment in cancer: molecular mechanisms and therapeutic interventions. *Signal Transduction and Targeted Therapy*, 8(1), February 2023. ISSN 2059-3635. doi: 10.1038/s41392-023-01332-8. URL http://dx.doi.org/10.1038/s41392-023-01332-8.

[31] Daniel Öhlund, Ela Elyada, and David Tuveson. Fibroblast heterogeneity in the cancer wound. *Journal of Experimental Medicine*, 211(8):1503–1523, July 2014. ISSN 0022-1007. doi: 10.1084/jem.20140692. URL http://dx.doi.org/10.1084/jem.20140692.

[32] Raghu Kalluri. The biology and function of fibroblasts in cancer. *Nature Reviews Cancer*, 16(9):582–598, August 2016. ISSN 1474-1768. doi: 10.1038/nrc.2016.73. URL http://dx.doi.org/10.1038/nrc.2016.73.

[33] Giulia Biffi and David A. Tuveson. Diversity and biology of cancer-associated fibroblasts. *Physiological Reviews*, 101(1):147–176, January 2021. ISSN 1522-1210. doi: 10.1152/physrev.00048.2019. URL http://dx.doi.org/10.1152/physrev.00048.2019.

[34] Jinfang Zhu and William E. Paul. Cd4 t cells: fates, functions, and faults. *Blood*, 112(5):1557–1569, September 2008. ISSN 1528-0020. doi: 10.1182/blood-2008-05-078154. URL http://dx.doi.org/10.1182/blood-2008-05-078154.

[35] E. John Wherry and Makoto Kurachi. Molecular and cellular insights into t cell exhaustion. *Nature Reviews Immunology*, 15(8):486–499, July 2015. ISSN 1474-1741. doi: 10.1038/nri3862. URL http://dx.doi.org/10.1038/nri3862.

[36] Susan M. Kaech and Weiguo Cui. Transcriptional control of effector and memory cd8+ t cell differentiation. *Nature Reviews Immunology*, 12(11):749–761, October 2012. ISSN 1474-1741. doi: 10.1038/nri3307. URL http://dx.doi.org/10.1038/nri3307.

[37] Grace X. Y. Zheng, Jessica M. Terry, Phillip Belgrader, Paul Ryvkin, Zachary W. Bent, Ryan Wilson, Solongo B. Ziraldo, Tobias D. Wheeler, Geoff P. McDermott, Junjie Zhu, Mark T. Gregory, Joe Shuga, Luz Montesclaros, Jason G. Underwood, Donald A. Masquelier, Stefanie Y. Nishimura, Michael Schnall-Levin, Paul W. Wyatt, Christopher M. Hindson, Rajiv Bharadwaj, Alexander Wong, Kevin D. Ness, Lan W. Beppu, H. Joachim Deeg, Christopher McFarland, Keith R. Loeb, William J. Valente, Nolan G. Ericson, Emily A. Stevens, Jerald P. Radich, Tarjei S. Mikkelsen, Benjamin J. Hindson, and Jason H. Bielas. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8(1), January 2017. ISSN 2041-1723. doi: 10.1038/ncomms14049. URL http://dx.doi.org/10.1038/ncomms14049.

[38] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631, 2019.

[39] Omry Yadan. Hydra - a framework for elegantly configuring complex applications. Github, 2019. URL https://github.com/facebookresearch/hydra.

A Appendix / supplemental material

A.1 Bootstrapped Graph Latents (BGRL)

Bootstrapped Graph Latents (BGRL) [10] is a self-supervised graph representation learning method used in this project. It avoids labels and negative samples by predicting alternate augmentations of the same input graph.

A graph $\mathbf{G} = (\mathbf{X}, \mathbf{A})$ is first augmented into two alternate views $\mathbf{G}_1 = (\tilde{\mathbf{X}}_1, \tilde{\mathbf{A}}_1)$ and $\mathbf{G}_2 = (\tilde{\mathbf{X}}_2, \tilde{\mathbf{A}}_2)$ via graph augmentation functions $\mathcal{T}_1$ and $\mathcal{T}_2$, respectively. An online encoder $\mathcal{E}_\theta$ with parameters θ then produces an online representation from the first augmented view, $\tilde{\mathbf{H}}_1 := \mathcal{E}_\theta(\tilde{\mathbf{X}}_1, \tilde{\mathbf{A}}_1)$, and a target encoder $\mathcal{E}_\phi$ with parameters ϕ produces a target representation from the second augmented view, $\tilde{\mathbf{H}}_2 := \mathcal{E}_\phi(\tilde{\mathbf{X}}_2, \tilde{\mathbf{A}}_2)$. A prediction of the target representation, $\tilde{\mathbf{Z}}_1 := p_\theta(\tilde{\mathbf{H}}_1)$, is obtained by feeding the online representation into a node-level predictor p_θ.

To update the online encoder's parameters θ, the gradient of the cosine similarity of the predicted target representation $\tilde{\mathbf{Z}}_1$ and the true target representation $\tilde{\mathbf{H}}_2$ is computed with respect to θ:

$$l(\theta, \phi) = -\frac{2}{N} \sum_{i=0}^{N-1} \frac{\tilde{\mathbf{Z}}_{(1,i)} \tilde{\mathbf{H}}_{(2,i)}^{\mathsf{T}}}{\left\|\tilde{\mathbf{Z}}_{(1,i)}\right\| \left\|\tilde{\mathbf{H}}_{(2,i)}\right\|} \tag{7}$$

$$\theta \leftarrow \text{optimize}(\theta, \eta, \partial_\theta l(\theta, \phi)). \tag{8}$$

Here, η is the learning rate and in practice, the loss is symmetrized by also predicting the target representation of the first view with the online representation of the second view.

The target encoder's parameters ϕ are updated as an exponentially moving average with decay rate τ of the online encoder's parameters θ:

$$\phi \leftarrow \tau\phi + (1 - \tau)\theta. \tag{9}$$

A.2 Deep Graph Contrastive Representation Learning (GRACE)

Deep Graph Contrastive Representation Learning (GRACE) [9] is a self-supervised method for unsupervised graph representation learning. Unlike methods relying on global readouts, GRACE directly contrasts node-level embeddings across two randomly corrupted views of the same graph.

Formally, given a graph $\mathbf{G} = (\mathbf{X}, \mathbf{A})$, GRACE generates two augmented views $\mathbf{G}_1 = (\tilde{\mathbf{X}}_1, \tilde{\mathbf{A}}_1)$ and $\mathbf{G}_2 = (\tilde{\mathbf{X}}_2, \tilde{\mathbf{A}}_2)$ by applying stochastic corruption functions $\mathcal{T}_1, \mathcal{T}_2$ to features and edges. Specifically, GRACE uses (i) *edge removal* with probability p_r and (ii) *feature masking* with probability p_m to generate diverse contexts.

A shared GNN encoder f_θ then computes node embeddings $\mathbf{U} = f_\theta(\tilde{\mathbf{X}}_1, \tilde{\mathbf{A}}_1)$ and $\mathbf{V} = f_\theta(\tilde{\mathbf{X}}_2, \tilde{\mathbf{A}}_2)$. For a node i, the embeddings $(\mathbf{u}_i, \mathbf{v}_i)$ from the two views form a positive pair, while embeddings from other nodes act as negatives. The similarity between two embeddings is estimated by a critic

$$\theta(\mathbf{u}, \mathbf{v}) = \frac{g(\mathbf{u})^\top g(\mathbf{v})}{\|g(\mathbf{u})\| \, \|g(\mathbf{v})\|}, \tag{10}$$

where $g(\cdot)$ is a two-layer projection head and the similarity is scaled by a temperature τ.

The contrastive loss for node i is defined as

$$\ell(\mathbf{u}_i, \mathbf{v}_i) = -\log \frac{\exp(\theta(\mathbf{u}_i, \mathbf{v}_i)/\tau)}{\exp(\theta(\mathbf{u}_i, \mathbf{v}_i)/\tau) + \sum_{k \neq i} \exp(\theta(\mathbf{u}_i, \mathbf{v}_k)/\tau) + \sum_{k \neq i} \exp(\theta(\mathbf{u}_i, \mathbf{u}_k)/\tau)}. \tag{11}$$

The final symmetric objective averages over all nodes:

$$J = \frac{1}{2N} \sum_{i=1}^{N} \left[\ell(\mathbf{u}_i, \mathbf{v}_i) + \ell(\mathbf{v}_i, \mathbf{u}_i)\right]. \tag{12}$$

A.3 Augmentation benchmark

To assess the computational costs associated with different augmentations and combinations of augmentations, they were applied to synthetic graphs of varying sizes while measuring runtime and memory usage.

For augmentations relevant to domain identification, synthetic graphs were generated to mimic the structure of real domain identification data. These graphs consisted of nodes with 50 numerical features, with feature similarities reflecting group structures, i.e., nodes within a group had more

similar features than those in different groups. For phenotype prediction augmentations, graphs were designed to contain nodes annotated with a cell type feature and a cell size feature. Additionally, edges were annotated with a binary indicator distinguishing "near" from "distant" connections.

All individual augmentations applicable to either domain identification or phenotype prediction were tested on their respective synthetic graph types. Furthermore, combinations of augmentations, corresponding to those evaluated in the main experiments, were also benchmarked. Each augmentation or combination was applied to synthetic graphs of increasing size, with each experiment repeated three times on a single GPU. For each run, both the runtime and peak GPU memory usage were recorded. The mean values across the three replicates were reported as the final result.

The results for domain identification augmentations are shown in Figure 2. Augmentation modes using *DropImportance* exhibit higher runtime compared to baseline augmentations (*DropFeatures* and *DropEdges*) and noise-based augmentations (*SpatialNoise* and *FeatureNoise*), though still running for 1 second or less for all graph sizes. Smoothing exhibits the highest memory usage of all the augmentations.
Note: The relatively high runtime observed for smaller graphs primarily reflects fixed computational overheads (e.g., data loading, graph construction, and GPU initialization), which dominate when per-graph computation is fast. These effects diminish as graph size increases, where runtime scales more proportionally with the number of nodes and edges.

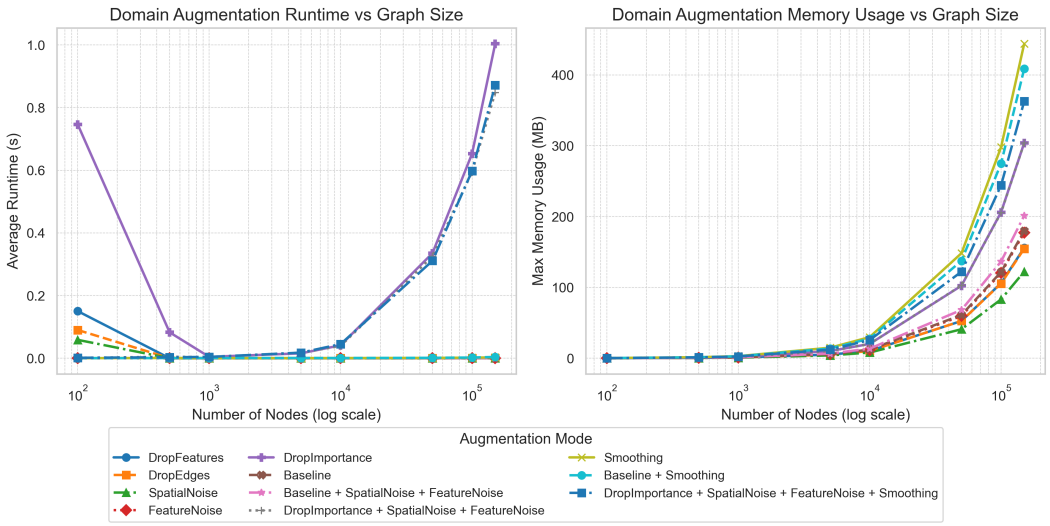

Figure 2: **Benchmark of domain identification augmentations.** Runtime (left) and peak GPU memory usage (right) for domain identification augmentations across increasing graph sizes. Each line represents either an individual augmentation or a combination of augmentations.

The results for phenotype prediction augmentations are shown in Figure 3. The runtime scaling trends are similar to those in the domain identification results. Augmentation modes using *DropImportance* scale worse than baseline and noise-based augmentations in both runtime and memory usage.

Overall, the benchmark highlights substantial variability in the computational efficiency of different augmentation strategies. Especially more complex augmentations, such as *DropImportance* and *Smoothing*, significantly increase runtime and memory consumption on large graphs, which also introduces considerable computational overhead during model training.

A.4 Classification metrics

To assess the performance of the phenotype prediction model, several binary classification metrics were used. These were computed from the predicted logits $\mathbf{z} \in \mathbb{R}^N$ and the ground truth binary labels $\mathbf{y} \in \{0, 1\}^N$ for all N samples.

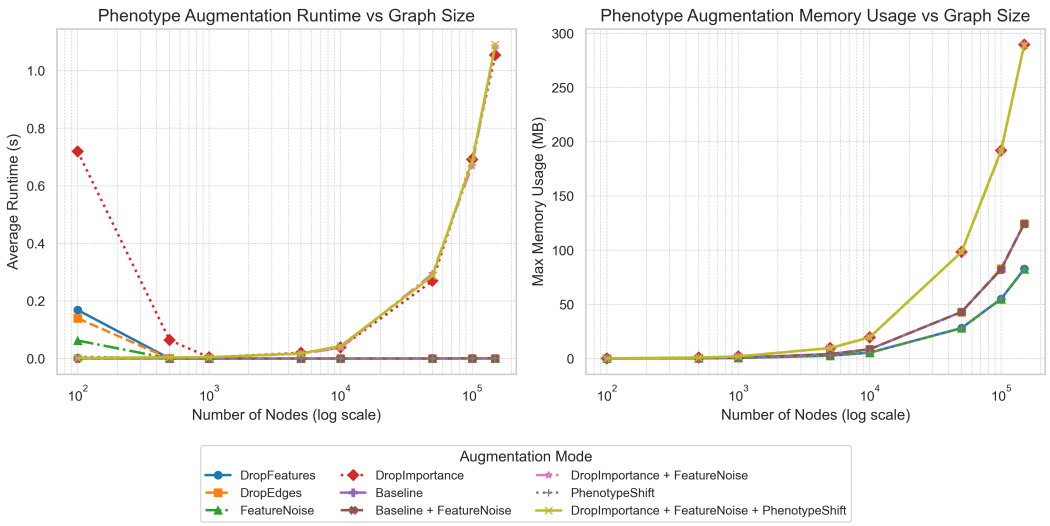

Figure 3: **Benchmark of phenotype prediction augmentations.** Runtime (left) and peak GPU memory usage (right) for phenotype prediction augmentations across increasing graph sizes. Each line represents either an individual augmentation or a combination of augmentations.

First, the predicted logits were transformed into probabilities using the sigmoid function:

$$\hat{\mathbf{p}} = \sigma(\mathbf{z}) = \frac{1}{1 + e^{-\mathbf{z}}} \tag{13}$$

A threshold $\tau \in [0, 1]$ was applied to convert probabilities into binary predictions:

$$\hat{\mathbf{y}} = \mathbb{I}[\hat{\mathbf{p}} \geq \tau] \tag{14}$$

During validation, the threshold τ was chosen to maximize the F1 score across a set of candidate thresholds. Once the optimal threshold was selected, the following metrics were computed:

- **AUROC (Area Under the Receiver Operating Characteristic Curve)**: The AUROC quantifies the probability that a randomly chosen positive sample is ranked higher than a randomly chosen negative sample by the model's scoring function. Formally, if $s(x)$ denotes the prediction score, then

  $$\text{AUROC} = \mathbb{P}\big(s(x^+) > s(x^-)\big)\,,$$

  where x^+ and x^- are independent draws from the positive and negative classes, respectively. Equivalently, AUROC corresponds to the area under the curve tracing the true positive rate (TPR) against the false positive rate (FPR) as the classification threshold is varied:

  $$\text{TPR}(t) = \frac{\text{TP}(t)}{\text{TP}(t) + \text{FN}(t)}, \quad \text{FPR}(t) = \frac{\text{FP}(t)}{\text{FP}(t) + \text{TN}(t)},$$

  where TP, FP, TN, FN denote true/false positives/negatives at threshold t. A value of 0.5 corresponds to random guessing, while 1.0 indicates perfect class separability.
- **Precision**: Fraction of predicted positives that are correct:

  $$\text{Precision} = \frac{TP}{TP + FP} \tag{15}$$

- **Recall (Sensitivity)**: Fraction of actual positives that are correctly identified:

  $$\text{Recall} = \frac{TP}{TP + FN} \tag{16}$$

- **F1 Score**: Harmonic mean of precision and recall, balancing both metrics:

  $$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{17}$$

A.5 Clustering evaluation metrics

To evaluate the quality of clustering results obtained, three metrics were employed: Normalized Mutual Information (NMI), Homogeneity, and Completeness. These metrics assess how well the predicted clustering aligns with ground truth domain labels.

NMI measures the mutual dependence between the predicted clustering C and the ground truth labels Y, normalized by the entropy of both. It is defined as:

$$\text{NMI}(C, Y) = \frac{2 \cdot I(C; Y)}{H(C) + H(Y)} \tag{18}$$

where $I(C; Y)$ is the mutual information between C and Y, and $H(\cdot)$ denotes entropy. Mutual information is given by:

$$I(C; Y) = \sum_{c \in C} \sum_{y \in Y} P(c, y) \log \left(\frac{P(c, y)}{P(c)P(y)} \right) \tag{19}$$

Here, $P(c, y)$ is the joint probability of a sample being in cluster c and class y, while $P(c)$ and $P(y)$ are the marginal probabilities.

Homogeneity assesses whether each cluster contains only data points that belong to a single class. It is defined as:

$$\text{HOM}(C, Y) = \begin{cases} 1 & \text{if } H(Y|C) = 0 \\ 1 - \frac{H(Y|C)}{H(Y)} & \text{otherwise} \end{cases} \tag{20}$$

where $H(Y|C)$ is the conditional entropy of the ground truth labels given the cluster assignments, and $H(Y)$ is the entropy of the ground truth.

Completeness measures whether all members of a given class are assigned to the same cluster. It is defined as:

$$\text{COM}(C, Y) = \begin{cases} 1 & \text{if } H(C|Y) = 0 \\ 1 - \frac{H(C|Y)}{H(C)} & \text{otherwise} \end{cases} \tag{21}$$

where $H(C|Y)$ is the conditional entropy of the predicted cluster assignments given the true class labels.

A.6 Possible cell type transitions for the *PhenotypeShift* augmentation

We allow a restricted set of biologically motivated cell type transitions, reflecting known plasticity and differentiation processes in the tumor microenvironment:

- **Tumor adaptation:** Tumor cells (normal) can transition to hypoxic tumor states [30].
- **Fibroblast (CAF) plasticity:** Collagen CAFs may become myofibroblastic CAFs (mCAFs) or adapt to hypoxia; mCAFs can further switch into SMA^+ CAFs, $PDPN^+$ CAFs, vascular CAFs, or hypoxic CAFs; iCAFs can adopt $PDPN^+$ or IDO^+ states; IDO^+ CAFs can also adapt to hypoxia; tumor-promoting CAFs (tCAFs) can transition to hypoxic tCAFs [31–33].
- **$CD4^+$ T cell differentiation:** CD4 T cells can give rise to regulatory T cells (Tregs), $PD1^+$ exhausted cells, IDO^+ subsets, proliferative ($Ki67^+$) states, or $TCF1/7^+$ progenitor-like cells [34, 35].
- **$CD8^+$ T cell differentiation:** CD8 T cells can give rise to IDO^+ subsets, proliferative ($Ki67^+$) states, or $TCF1/7^+$ progenitor exhausted cells [35, 36].
- **Myeloid refinement:** Myeloid cells can be further refined into neutrophil identities, reflecting annotation resolution rather than a true biological transition [37].

A.7 Hyperparameter Search.

We conducted automated hyperparameter optimization using Optuna [38] with the Hydra [39] sweeper integration. For the domain identification task, we ran 100 trials per configuration, optimizing for validation NMI. For the phenotype prediction task, we ran 10 trials, optimizing for validation F1.

Both model and augmentation hyperparameters were tuned jointly, ensuring fair comparisons across different augmentation regimes. All searches were performed exclusively on validation splits, with the test set kept untouched until final evaluation. Each experiment was repeated with 5 random seeds, and reported means and standard deviations capture variability across seeds rather than across search trials. To ensure fairness, all augmentation regimes were allocated identical search budgets and evaluated under the same conditions.

All searches were executed on ETH's LeoMed cluster using NVIDIA RTX 4090 GPUs (24GB) with 6–16 CPU cores and 16–96 GB RAM, depending on the task. Domain identification runs completed within approximately 30 minutes per trial, while phenotype prediction runs required up to 4 hours per trial.

A.8 Hyperparameter ranges used for tuning augmentations

Table 5: **Hyperparameter search ranges for graph augmentations.** For each augmentation, the tuned hyperparameters and their respective ranges are listed. Intervals denote uniform sampling from the specified range.

Augmentation	**Hyperparameter**	**Range**
DropEdges	p	$[0.1, 0.4]$
DropFeatures	p	$[0.1, 0.4]$
DropImportance	λ_p μ	$[0.4, 0.6]$ $[0.1, 0.4]$
SpatialNoise	σ_{spatial}	$[2.0, 30.0]$
FeatureNoise	σ_{feature}	$[0.05, 1.0]$
SmoothFeatures	α	$[0.0, 0.5]$
PhenotypeShift	p_{shift}	$[0.0, 0.3]$

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: The abstract and introduction clearly state the scope and contributions of the paper. They emphasize that the work systematically investigates graph augmentations for self-supervised learning in spatial omics, introduces both existing and novel biologically motivated augmentations, and evaluates them on two representative tasks (domain identification and phenotype prediction).

   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: The paper includes a dedicated discussion of limitations. It acknowledges that the study is restricted to two downstream tasks (domain identification and phenotype prediction), with domain identification relying on a small number of annotated healthy samples and phenotype prediction limited to a single cancer cohort.

   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: [NA]

   Justification: The paper includes no theoretical results.

   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: The paper provides all necessary details to reproduce the main experimental results. The architectures, loss functions, preprocessing pipelines, and training procedures are fully described. Hyperparameter search spaces are reported in the appendix for completeness.

   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: All datasets used are publicly available with citations to the original sources. Code is provided as Supplementary Material.

   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: The paper specifies dataset splits, preprocessing steps, model architectures, training procedures, optimizers, and hyperparameter search ranges, with full details included in the appendix for reproducibility.

   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: The paper reports mean metrics and 1-sigma error bars calculated over 5 runs of each experiment using different random seeds. This is indicated in the main text of the paper and table captions themselves,

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: All compute resources used are indicated in the appendix of the paper.

   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: The work relies solely on publicly available datasets and complies with ethical standards for data usage and research integrity.

   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: The paper discusses potential positive impacts, such as advancing spatial omics analysis for biomedical research and clinical application.

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: The 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: All datasets are publicly available and properly cited in the paper. Where licenses or terms of use are specified (e.g., CC-BY for NSCLC IMC data), they were followed; where not explicitly specified, datasets were used in line with the original publications' terms for academic research.

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: [Yes]

    Justification: The provided code is properly documented and the documentation is provided in the code repository in Supplementary Materials.

    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: The 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: [NA]

Justification: The 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.