Biological network inference from single-cell multi-omics data using heterogeneous graph transformer

Anjun Ma[1,2,*], Xiaoying Wang[3,*], Cankun Wang[1], Jingxian Li[3], Tong Xiao[2], Juexing Wang[4], Yuzhou Chang[1,2], Yang Li[1], Yutao Liu[3], Shaopeng Gu[1], Duolin Wang[4], Yuexu Jiang[4], Jinpu Li[4], Li Su[4], Zihai Li[2], Bingqiang Liu[3,$], Dong Xu[4,$], Qin Ma[1,2,$]

[1] Department of Biomedical Informatics, College of Medicine, The Ohio State University, Columbus, OH, 43210, USA,
[2] Pelotonia Institute for Immuno-Oncology, The Ohio State University Comprehensive Cancer Center, Columbus, OH, 43210, USA,
[3] School of Mathematics, Shandong University, Jinan, Shandong, 250100, China,
[4] Department of Electrical Engineering and Computer Science, and Christopher S. Bond Life Sciences Center, University of Missouri, Columbia, MO 65211, USA.

* These authors contributed equally to the paper as first authors
[$] To whom correspondence should be addressed

Abstract

We present DeepMAPS, a deep learning platform for cell-type-specific biological gene network inference from single-cell multi-omics (scMulti-omics). DeepMAPS includes both cells and genes in a heterogeneous graph to infer cell-cell, cell-gene, and gene-gene relations simultaneously. The graph attention neural network considers a cell and a gene with both local and global information, making DeepMAPS more robust to data noises. We benchmarked DeepMAPS on 18 datasets for cell clustering and network inference, and the results showed that our method outperforms various existing tools. We further applied DeepMAPS on a case study of lung tumor leukocyte CITE-seq data and observed superior performance in cell clustering, and predicted biologically meaningful cell-cell communication pathways based on the inferred gene networks. To improve the feasibility and ensure the reproducibility of analyzing scMulti-omics data, we deployed a webserver with multi-functions and various visualizations. Overall, we valued DeepMAPS as a novel platform of the state-of-the-art deep learning model in the single-cell study and can promote the use of scMulti-omics data in the community.

Keywords: Single cell multimodal omics, heterogeneous graph transformer, graph neural network, multi-head attention framework, biological network, single-cell sequencing data analysis webserver

Main

Single-cell sequencing, such as single-cell RNA sequencing (scRNA-seq) and single-cell ATAC sequencing (scATAC-seq), reshapes the way of investigating cellular heterogeneity and brings novel insights in neuroscience, cancer research, immuno-oncology, drug response, etc[1, 2]. Individual single-cell modality can only reflect the snapshot of genetic features and partially depict the peculiarity of cells, leading to characterization biases in complex biological systems[2, 3]. To fully capture the intricacy of complex molecular mechanism and cellular heterogeneity, single-cell multi-omics (scMulti-omics) generate and quantify multiple modalities in the single-cell level simultaneously, and such measurements advance various biological studies equipped with robust computational analysis methods[4].

The existing tools for integrative analyses of scMulti-omics data, e.g., Seurat[5], MOFA+[6], Harmony[7], and totalVI[8], achieve reliable prediction of cell types and states, recover dropouts in single-modality data analysis, remove batch effects, reveal relationships or alignment among various modalities, and toward a mechanistic understanding of cell-type-specific gene regulations. However, the cross-talk among cells and different molecular modalities (e.g., genes and proteins) are usually missing in an independent hypothesis. Hence, these models lack the ability to infer the underlying biological networks of diverse cell types and have limited power to elucidate the response of these complex networks to external stimuli in specific cell types.

Recently, graph neural network (GNN) shows its unique strength in learning low-dimensional representations of individual cells by propagating neighbor cells' features and constructing cell-cell relations in a global cell graph[9-12]. For example, our in-house tool scGNN is a novel GNN model that has demonstrated the superior performance of cell clustering and gene imputation based on the large-scale scRNA-seq data[13]. Furthermore, a heterogeneous graph is a multi-relational model which provides a natural representation framework for integrating scMulti-omics data and learning the underlying cell-type-specific biological networks. Moreover, the recent development in the attention mechanism for modeling and integrating heterogeneous relationships make deep learning models explainable and enable the inference of cell-type-specific biological networks[14, 15].

To this end, we develop the first-of-its-kind model **DeepMAPS** (**Deep** learning-based **M**ulti-omics **A**nalysis **P**latform for **S**ingle-cell data), which is a heterogeneous graph transformer framework for biological network inference from scMulti-omics data. DeepMAPS formulates high-level representations of relations among cells and genes in a heterogeneous graph, with cells and genes as the two disjoint node sets in this graph. Projecting the features of genes and cells into the same latent space is an effective way to harmonize the imbalance of different batches and lies a solid foundation of cell clustering (i.e., node clustering) and the prediction of cell-gene and gene-gene relations

in a specific cell cluster (i.e., link prediction). Most importantly, the attention mechanism in this transformer model enhances the biological interpretability and enables the identification of important gene modules in each cell cluster. Overall, DeepMAPS is an end-to-end and hypotheses-free framework and provides the first deep learning tool to infer the cell-type-specific biological networks from scMulti-omics data.

In this study, we evaluated DeepMAPS on 18 scMulti-omics datasets (including transcriptomic, epigenomic, and proteomic data) and our method outperformed existing tools in terms of cell clustering accuracy and biological network inference (i.e., gene association network and gene regulatory network). Specifically, DeepMAPS shows its superior power in characterizing cellular heterogeneity since it can pass messages of neighbor cells and genes to employ the identification of highly important genes in each cell cluster. To further validate the biological insight inferred from DeepMAPS, we performed a case study on a lung tumor leukocyte CITE-seq data. By jointly analyzing gene expression and protein abundance, DeepMAPS accurately identified and annotated 13 cell types based on curated markers, which cannot be fully elucidated by a single modality. We also proved that the embedding features identified in DeepMAPS capture the true signals and amplify them when the original signals are weak. Besides, we also identified biologically meaningful cell-cell communication pathways between dendritic cells and tissue resident memory CD4 T cells based on the gene network inferred in the two clusters. We deployed DeepMAPS as a code-free web portal along with Dockers, to ensure the reproducibility of scMulti-omics data analysis and lessen the programming burden for biologists who lack sufficient computational skills or resources (**Fig. 1**).

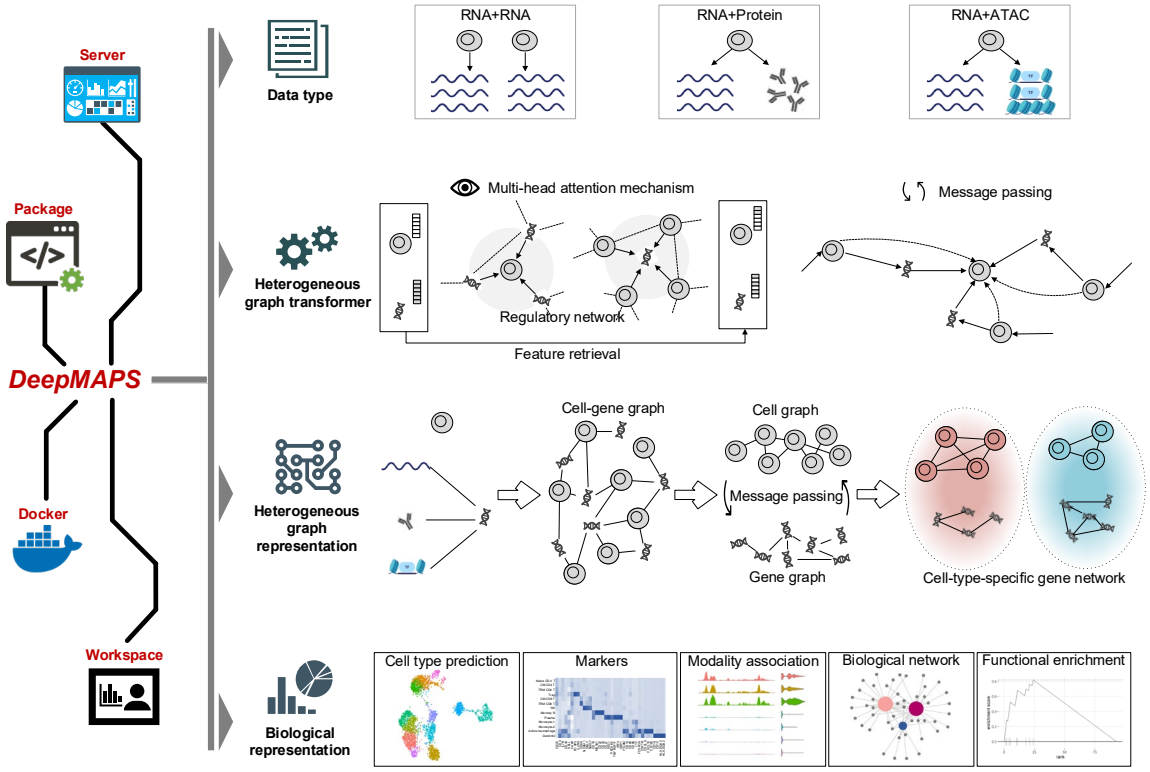

Fig. 1. DeepMAPS is a deep learning-based Multi-omics Analysis Platform for Single-cell data. It allows the joint analysis of multiple scRNA-seq, CITE-seq, and matched single-cell RNA and ATAC-seq data. The core method includes the representation of cell-gene relations via a heterogeneous graph and a transformer with a graph attention mechanism. DeepMAPS provides interactive and interpretable graphical representations to deliver cell clusters and various cell-type-specific biological networks depending on modality types. Eventually, DeepMAPS is delivered as a web portal to ensure robustness and reproducibility, along with a docker container. Workspace is committed to being provided for users for job saving and retrieval. DeepMAPS also supports diverse interpretations include but not limited to joint cell clustering, marker identification, modality associations, cell-type-specific biological network inference, and functional enrichment.

Results

Overview of DeepMAPS

There are five steps in DeepMAPS to fulfill the joint analysis of scMulti-omics data (**Fig. 2** and **Methods**). (*i*) Data is preprocessed by removing low-quality cells and modalities and then applied with different normalization methods according to specific data types (*ii*) An integrated cell-gene matrix will be generated by representing the combined activity of each gene in each cell. Different data integration methods are applied for different scMulti-omics data types. (*iii*) A heterogeneous graph transformer (**HGT**) model is built to jointly learn the low-dimensional embedding for cells and genes and generate an attention score to indicate the importance of a gene to a cell. (*iv*) Cell clustering and identification of functional gene modules in each cell cluster based on HGT-learned embeddings and

attention score. (*v*) Diverse biological networks, e.g., gene regulatory networks and gene association networks, are inferred in each cell type.

To enable the simultaneous learning of joint representatives of cells and genes, we first generate a cell-gene matrix integrating all heterogeneous information of the input scMulti-omics data. A heterogeneous graph with cell nodes and gene nodes is then constructed, where a cell-gene edge represents the integrated gene activity score in the matrix, and the initial embedding of each node is learned from the gene-cell integrated matrix via a two-layer GNN graph autoencoder. The entire heterogeneous graph is then sent to a graph autoencoder to learn relations among cells and genes and update the embedding of each node. Here, DeepMAPS adopts a heterogeneous multi-head attention mechanism to models both overall topological information (global relationship) and neighbor message passing (local relationship) on the heterogeneous graph.

In a HGT layer, each node (either a cell or a gene) will be considered as a target, and DeepMAPS evaluates the importance of its neighbor nodes and the amount of information that can be passed to the target based on the synergy of node embedding. As a result, cells and genes with high positively correlated embeddings are more likely to exchange messages with each other, thus tending to maximize the similarity and disagreement of embeddings. To make the unsupervised training process feasible on a large heterogeneous graph, DeepMAPS is first performed on 50 subgraphs subtracted from the heterogeneous graph, covering a minimum coverage of 50% of all nodes, to train a shared parameters between different nodes, which is later been used for testing the whole graph. As an important training outcome, an attention score will be given to represent the importance of a gene to a cell. A gene with high attention to a cell implies that the gene is of relatively much importance for defining cell identity and characterizing cell heterogeneity. This insight will lead to reliable gene association networks in each cell cluster as the final output of DeepMAPS.

We applied the graph-based invariant-sampling method to make DeepMAPS feasible for handling large-scale heterogeneous graphs[16]. The basic idea is to keep a separate node budget for each node type (i.e., cell node and gene node) and to sample an equal number of nodes per type with a sampling strategy to reduce variance. For an already sampled node, we will add all its direct neighbors into the corresponding budget and add its normalized degree to these neighbors, which will then be used to calculate the sampling probability. Such normalization is equivalent to accumulating the random walk probability of each sampled node to its neighborhood, avoiding the sampling being dominated by high-degree nodes. The sampler constructs a number of small subgraphs from the given giant graph, and these subgraphs can be fed in batches with multiple GPUs. These distributed training results will be collected to build the whole graph with representations on the head node.

The gene-gene and gene-cell relations predicted via the HGT model can only depict the co-expression correlations as well as the gene activity in cells in the dataset rather than reflect the gene module specificity in a cell cluster. Therefore, we build a Steiner Forest Problem (SFP) model on a sparse heterogeneous graph to identify genes with higher attention scores and similar embedding features uniquely in a cell cluster. We firstly construct a sparse weighted heterogeneous graph based on the gene-gene and gene-cell relations, and then build an SFP model based on this heterogeneous graph and cell clusters. The gene-gene and gene-cell relations in the optimized solution of the SFP model mirror the co-expression relations among genes and the specificity of genes to a cell type. A gene network established from SFP contains genes that are highly associated based on their gene activities and are of the most important in characterizing the identity of that cell cluster, which is considered to be cell-type-active.

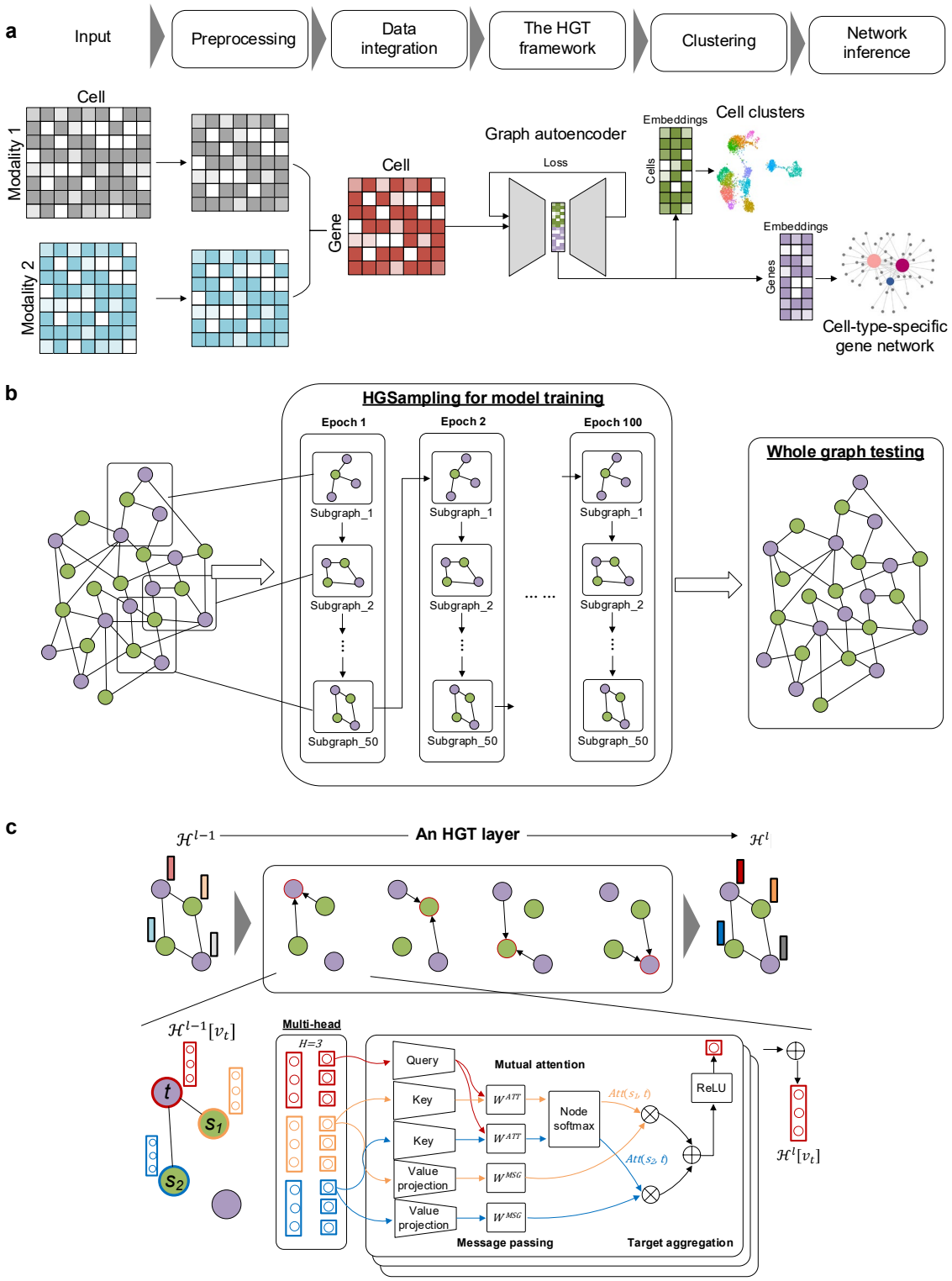

Fig. 2: DeepMAPS framework. (**a**) The five major steps and graphical illustration in DeepMAPS. (**b**) Relations in the heterogeneous graph of cells and genes will be learned in an HGT-based graph autoencoder. The hyperparameters are first trained in 50 subgraphs and 100 epochs and then applied to the testing in the whole graph. (**c**) For each subgraph training and the whole graph testing,

multiple HGT layers are applied. A toy example with two cells (purple) and two genes (green) are shown for one HGT layer. The embeddings of the target cell (red) and two neighbor genes (orange and blue) are separated evenly into three heads. For each head, the HGT layer calculates the attention score of neighbor genes to the target cell and updates the target's embedding of this head. The entire target cell embedding is updated by concatenating all three heads to complete one HGT layer.

DeepMAPS achieves superior performances in joint cell clustering and biological network inference from scMulti-omics data

We collected 18 scMulti-omics datasets, including two multiple scRNA-seq data (Data 1-2), eight CITE-seq data (Data 3-10), and eight matched scRNA-seq and scATAC-seq (scRNA-ATAC-seq) data measured from the same cell (Data 11-18), to benchmark DeepMAPS (**Supplementary Table S1**). Specifically, Data 1-2 and 17-18 have benchmark annotations provided in the original manuscript. These data cover a number of cells ranging from 549 to 30,672; an average read depth (consider scRNA-seq data only) ranging from 2,933 to 645,526; a zero-expression rate (consider scRNA-seq data only) from 71% to 97% (**Fig. 3a**).

To evaluate the performance of joint cell clustering in all three scMulti-omics data types, we compare DeepMAPS with four benchmarking tools using default settings, including Seurat, MOFA+, TotalVI, and Harmony (**Methods**), in terms of the Average Silhouette Weight (ASW) (**Fig. 3a**) and Adjusted Rand Index (ARI) (**Fig. 3b**). DeepMAPS was trained by each scMulti-omics data type and each dataset in an unsupervised way. The one set of parameter was chosen as default for all datasets in the same data type based on the grid optimization of hyper-parameter combinations (**Supplementary Table 3-5**). Results clearly showed that, in all three scenarios of scMulti-omics data, DeepMAPS distinctly outperformed the others in most cases. Note that, for Data 2 and 18, though DeepMAPS did not achieve the best ASW than other tools, its performance regarding ARI comparison to the benchmark label is the highest.

We showed UMAP results of cell clustering of Data 17, a cancer cell line data (n=549) with benchmarked cell labels of scRNA-ATAC-seq data (**Fig. 3c**). By comparing with the original cell line labels, we found that DeepMAPS is the only tool that accurately separate each cell type with minimum mismatches (ARI=0.97), while for Seurat (ARI=0.88) and MOFA+ (ARI=0.79), either PDX1 or PDX2 population was mistakenly divided into two clusters and include more mismatches. For datasets without benchmark labels, DeepMAPS also showed good UMAP visualization than other tools without having mixture clusters or separated random cells (**Fig. 3d)**.

To evaluate the robustness of DeepMAPS, we performed a leave-out test on benchmark datasets. For data with benchmark labels, we first filter cells by removing a cluster of cells

based on benchmark labels and then perform analysis using DeepMAPS and benchmark tools; for other datasets, we removed cells based on clusters identified from each tool. The results showed that DeepMAPS achieved superior performance compared to all other benchmarking methods (**Fig. 3e)**. Another test was performed on a series of data simulated for different read depth rates (**Fig. 3f**). For each testing dataset, the clustering results of DeepMAPS are consistent with high AWS or ARI, indicating that the message passing and attention mechanism used in DeepMAPS helped maintain cell-cell relations and tolerance to read depth.

We further evaluated the two kinds of biological networks the DeepMAPS can deliver. For the gene association network (for all scMulti-omics data types), we considered using the centrality scores and enriched pathways to compare DeepMAPS with IRIS3. IRIS3 is an in-house tool for the identification of cell-type-specific regulon and gene regulatory network construction from scRNA-seq data. It has superior performance than other public tools, such as SCENIC. We also compared our results to the co-expression network constructed from the whole gene list with significance cutoffs, which is a widely used way in single-cell studies. Both the average closeness centrality and eigenvector centrality scores of networks constructed in DeepMAPS showed significantly higher scores than the other two methods using all 18 benchmark datasets (**Fig. 3g-h**). Moreover, for the gene regulatory network constructed from scRNA-ATAC-seq data, we evaluated the number of significantly enriched pathways in a TF-regulon and in a cell-type-specific regulon (**Fig. 3i-j**). The results indicated that DeepMAPS is capable of constructing more compatible and biologically reasonable gene networks in each cell type and outperformed the other methods.

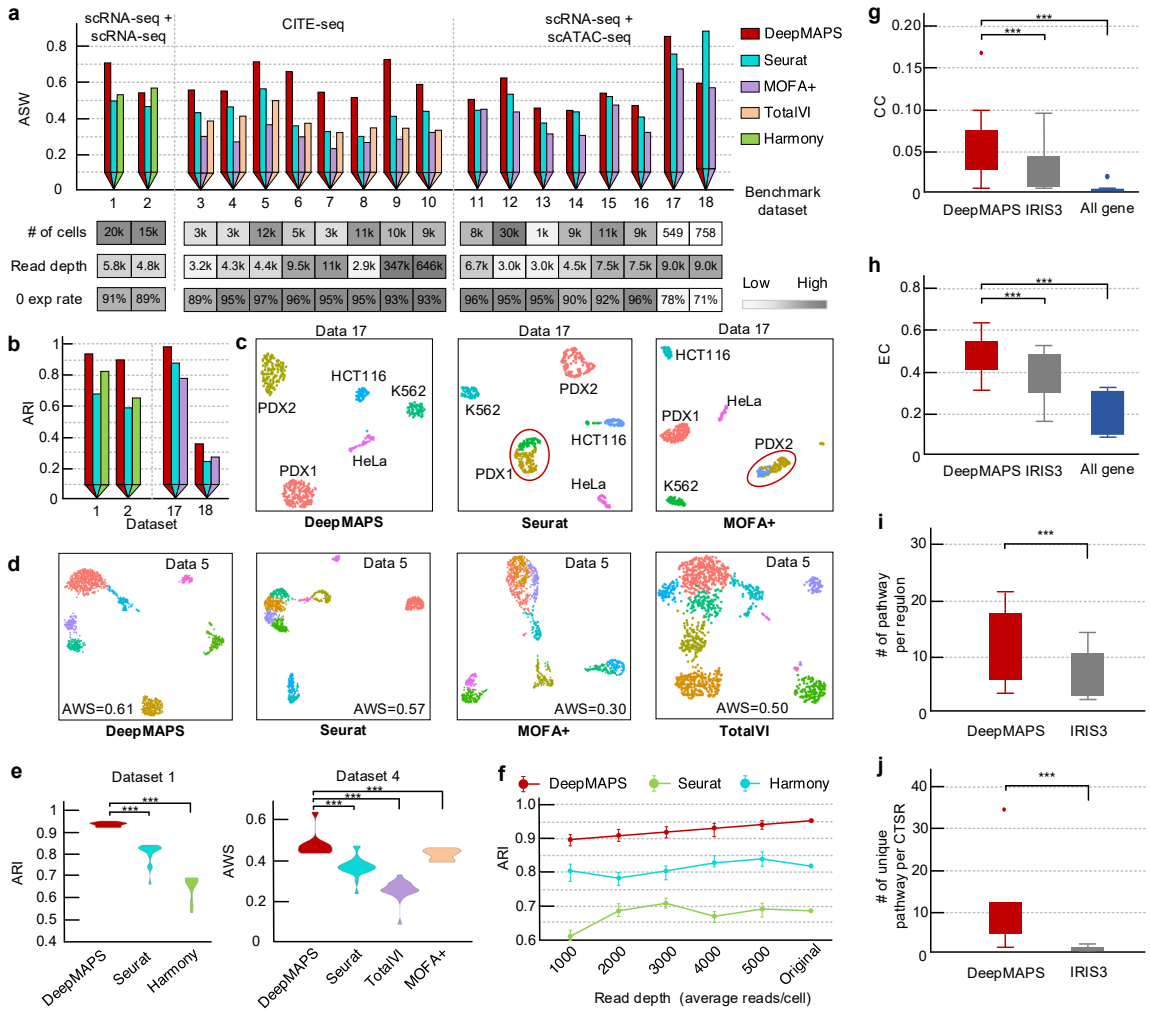

Fig. 3: Benchmarking of DeepMAPS in terms of cell clustering and biological network inference. (**a**) Benchmarking cell clustering results in all 18 datasets in terms of average silhouette weight without using benchmark cell labels. Due to the capability, different benchmarking tools were selected for the comparison. We also show heatmaps to indicate the number of total cells, average gene expression read depth per cell, and average RNA zero expression rate in each data. (**b**) Results comparison on four datasets with benchmarking cell labels in terms of adjusted rand index. (**c**) UMAP comparison of Data 17 (with benchmark labels) between DeepMAPS and other tools. Cluster labels were annotated based on cell correspondence to the original cell label. (**d**) UMAP comparison of Data 5 (without benchmark labels) between DeepMAPS and other tools. (**e**) Robustness test of DeepMAPS using cell cluster leave-out method for Data 1 (with benchmark label) and Data 4 (without benchmark label). Details can be found in the Method section. (**f**) Robustness test of DeepMAPS to different read depth on Data 1. (**g-h**) Evaluation and comparison of gene association network inference of DeepMAPS and other methods. Closeness centrality and betweenness centrality were used to indicate the compactness and connectivity of networks inferred from different methods. (**i-j**) Evaluation and comparison of gene regulatory network

(regulons) identified in DeepMAPS and IRIS3, based on the number of functional pathways enriched in a regulon or cell-type-specific regulon.

DeepMAPS accurately identify cell types in PBMC and lung tumor immune CITE-seq data

To demonstrate the joint representation of scMulti-omics in characterize cell identities, we demonstrate a case study that applies DeepMAPS on a published PBMC and lung tumor leukocytes CITE-seq data. The dataset includes RNA and protein measured on 3,485 cells. We identified 13 cell clusters, including four CD4 T cell groups (Naïve, central memory (CM), tissue resident memory (TRM), and regulatory (Treg)), two CD8 T cell groups (CM and TRM), a natural killer group, a memory B cell group, a plasma cell group, two monocyte groups, one tumor-associated macrophage group, and a dendritic cell (DC) group, via DeepMAPS and annotated each cluster by visualizing expression levels of curated maker genes and proteins (**Fig. 4a**). Compared to cell types identified using only proteins or RNAs, we either isolated or accurately annotated cell populations that cannot be characterized in the individual modality analysis. For example, the expression levels of marker genes (CD4, IL7R, CD44, SELL, and CD69) in CD4 T cells are weak and undistinguishable, thus only CM and naïve CD4 T cell groups can be recognized using only scRNA-seq data (**Fig. 4b**), while the signal of marker proteins in CD4 T cells are much stronger and differentiable. Another example lies in the recognition of tumor associated macrophage in which the marker genes (CSF1R, CCR2, and MARCO) were identified to be uniquely expressed in RNA, while no specific protein markers can be used to annotate such cell group. Altogether, combining signals captured from both RNA and protein sides, DeepMAPS successfully identified biologically reasonable and meaningful cell types in the CITE-seq data.

We then zoomed in to compare the modality correlation between two cell types. We used the top differentially expressed genes and proteins between memory B cells and plasma cells and performed hierarchical clustering of the correlation matrix. The result clearly stratified these features into two anticorrelated modules: one associated with memory B cells and the other with plasma cells (**Fig. 4c**). Furthermore, we found that the features in the two modules significantly correlated with the axis of maturation captured by our HGT embeddings. We observed that the 51th HGT embedding showed distinctive differences between plasma cells and memory B cells (**Fig. 4d-e**). Similar findings were also observed for the comparison of EM CD8 T cells and TRM CD8 T cells, which showed a much closer relation when looking at expression correlations (**Fig. 4f**). Nevertheless, we can still find a representative HGT embedding (56th) that maintains embedding signals to well separate the two groups (**Fig. 4g-h**). These results point to a program of any two cell clusters consisting of coordinated activation and repression of multiple genes and proteins, leading to a gradual transition in cell state that can be captured by a specific dimension of the DeepMAPS latent HGT space.

Based on the cell types and raw data of genes and proteins' expression, we inferred cell-cell communication by using CellChat[17]. We constructed communication networks among different cell types within multiple signaling pathways as **Fig. 4i.** We further applied CellChat[17] to find any ligand-receptor interactions that have been validated between any cell types we identified. We observed an ALCAM signaling pathway existing between TRM CD4 T cells and DCs in the lung cancer tumor microenvironment (TME), of which DCs and TRM CD4 T cells serve as the major source and ligand of CD166. Previous studies showed that ALCAM on antigen-presenting DCs would interact with CD6 on T cell surface and contribute to T-cell activation and proliferation[18-20]. For another example, we also identified the involvement of NECTIN pathway during the interaction between the activated macrophages (source) and TRM CD8 T cells (target). This is consistent with the previous report that NECTIN (CD155) expressed on tumor-infiltrating macrophages could be immunosuppressive when interacting with surface receptors on CD8+ T cells in the lung cancer TME[21, 22].

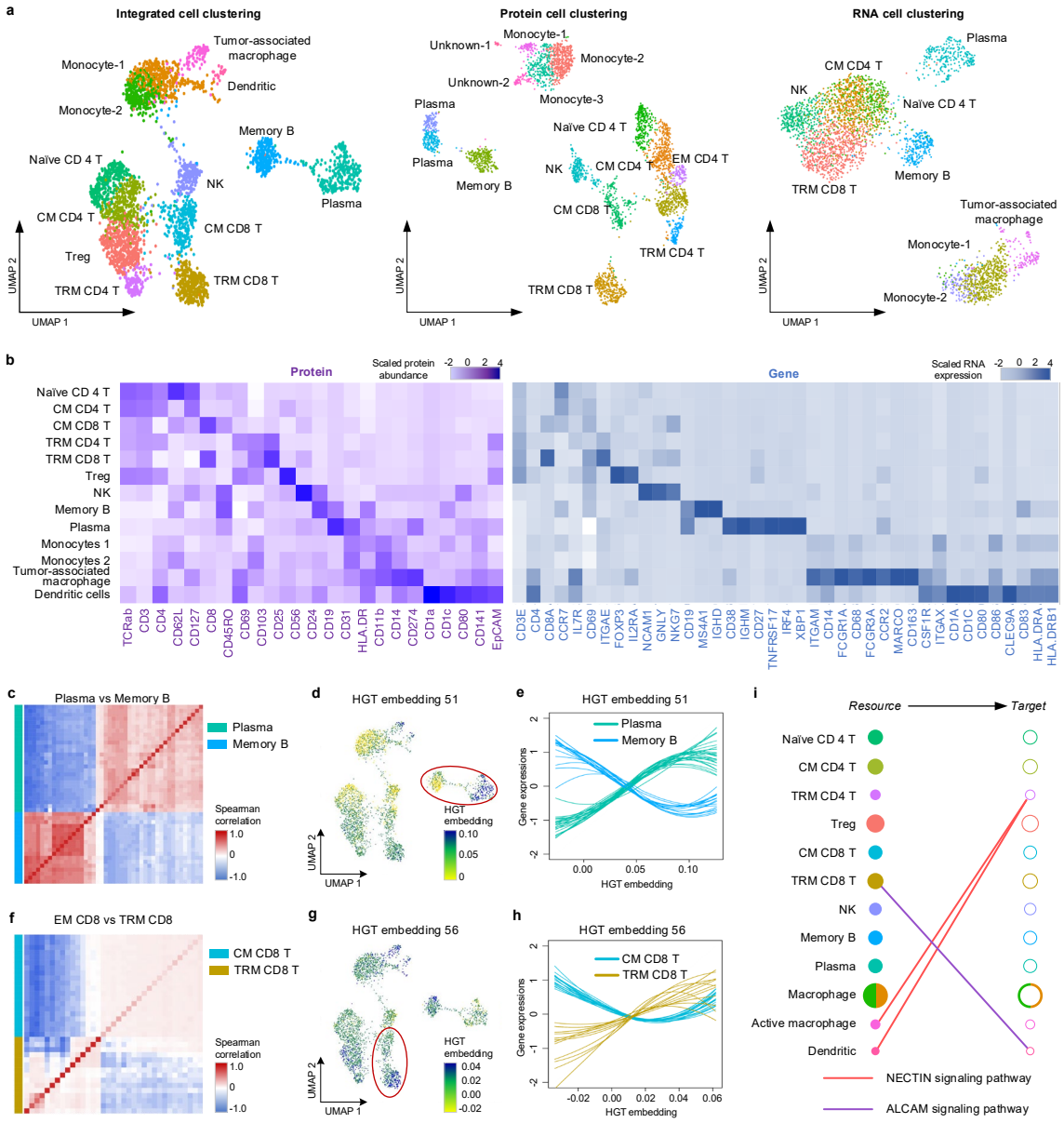

Fig. 4: DeepMAPS identifies the heterogeneity in CITE-seq data of PBMC and lung tumor leukocytes. (**a**) UMAPs for DeepMAPS cell clustering results of integrating RNA and protein information, only using protein, and only using RNA. Cell clusters were annotated based on curated marker proteins and genes. (**b**) Heatmap of curated marker proteins and genes that determine the cell clustering and annotation. (**c**) Heatmap of correlation comparison of top differentially expressed genes and proteins in plasma cells and memory B cells. (**d**) UMAP is colored by the 51th embedding, indicating distinct embedding representations in plasma cells and memory B cells. (**e**) Expression of top differentially expressed genes and proteins in ***c*** as a function of the 51th embedding to observe the pattern relations between plasma cells and memory B cells. Each line represents a gene/protein, colored by cell types. For each gene, the line was drawn by a loess smoothing function based on the corresponding embedding and scaled gene expression in a cell. (**f-h**) Similar

visualization was given for the comparison of EM CD8 T cell s and TRM cd8 T cells in ***c-e***. (**i**) Two signaling pathways, NECTIN and ALCAM, were shown to indicate the predicted cell-cell communications between two cell clusters.

DeepMAPS provides a multi-functional and user-friendly web portal for analyzing scMulti-omics data

Researchers who lack sufficient computational skills prefer to use webservers or dockers to lessen the programming burden of data analysis, and hence, a code-free and interactive platform for single-cell sequencing data analysis are urgently needed in the public domain. Considering the complexity of single-cell sequencing data, more and more webservers and dockers have been developed in the past three years[23-35] (**Supplementary Table 6**). However, most of these tools only provide minimal and basic functions such as cell clustering and differential gene analysis, and do not support the joint analysis of scMulti-omics data, especially lack sufficient support for biological network inference. To this end, we deliver DeepMAPS as the-first-of-its kind web portal to support online and code-free computational analysis for scMulti-omics data. The webserver supports the analysis of multiple RNA-seq data, CITE-seq data, and scRNA-ATAC-seq data using DeepMAPS. Other methods, e.g., Seurat, are also included as an alternative use for the users' convenience. Three major steps, data preprocessing, cell clustering and annotation, and network construction, are included in the server. The DeepMAPS server supports real-time computing and interactive graph representations. Users may register for an account to have their own workspace to store and share analytical results.

Conclusion and discussion

We highlighted DeepMAPS as the first deep learning framework that implements heterogeneous graph representation learning and graph transformer in the study of scMulti-omics data. By building a heterogeneous graph containing both cells and genes, DeepMAPS identified the joint embedding of both sides simultaneously and enabled the inference of cell-type-specific biological networks along with cell types in an intact framework. The application of heterogeneous graph transformer takes the advantages beyond graph nerural network that considering not only the message from neighbor cells but also the attention of how such message should be passed to the target. In such a way, the information training and learning process in a graph can be largely shortened to consider cell impacts from a further distance.

While with the advantages and outperformed performances in analyzing scMulti-omics data, there are still rooms to further improve the power of DeepMAPS. First of all, the computational efficiency for super large datasets (e.g., more than 1 million cells) might be a practical issue considering the complexity of the heterogeneous graph representation (may contain billions of edges). Moreover, DeepMAPS is recommended to be run on GPUs and supercomputers, which leads to a potential problem of reproducibility. Different

GPU models have different floating-point numbers that may influence the precision of loss functions during the training process. That is to say, for different GPU models, DeepMAPS may generate different cell clustering and network results, which is the main reason that drives us to the development of webserver. Lastly, the current version of DeepMAPS is based on a bipartite heterogeneous graph with genes and cells. Separate preprocessing and integration steps are required to transfer different modalities all into genes and be integrated into a unique cell-gene matrix. To fully achieve an end-to-end framework for scMulti-omics analysis, the bipartite graph can be extended to a multipartite graph, where each modality can be included as a node type. Such multipartite heterogeneous graph can even include knowledgeable and biological information, such as known molecular regulations, and more than two modalities all in one graph. However, by including more node types, the computational burden will be increased geometrically, which requires a dedicated discovery of model optimization in the future.

In summary, we evaluate our DeepMAPS as a pioneer study for the joint analysis of scMulti-omics data and cell-type-specific biological network inference. It is likely to provide new visions of deep learning deployment in single-cell biology. With the development and maintainness of DeepMAPS webserver, our long-term goal is to create a deep learning-based eco-community for AI-ready scMulti-omics data archiving, analyzing, visualizing, and disseminating.

Methods

Data preprocessing and data integration

Multiple scRNA-seq data

DeepMaps takes the raw scRNA-seq gene expression profiles as input. Only genes that are expressed in more than 0.1% of total cells, and cells with a minimum of 0.1% genes expressed were kept. To integrate multiple scRNA-seq, we first reduce the dimension of multiple scRNA-seq in low-dimensional space by canonical correlation analysis (CCA) and search for mutual nearest neighbors (MNNs) in the shared low-dimensional space (integration method used in Seurat v3). It then calculated vector of each cell and for correcting gene expressions in different datasets. The output is an integrated matrix with combined cells from all datasets and shared genes with normalized and scaled expression values of x'_{ij} for gene *i* in cell *j*.

CITE-seq data

We also first removed low-quality genes and cells as described above. We applied log normalization on the RNA matrix and centered log-ratio (CLR) transformation on the protein matrix as below:

$$Lognormalize(x_{ij}) = \log\left(1 + \frac{x_{ij} \times 1e4}{\|\{x_{i}.\}\|}\right) \quad (1)$$

$$CLRnormalize(x_{ij}) = \log\left(1 + \frac{x_{ij}}{L_j}\right) \quad (2)$$

where x_{ij} represent for the expression of either a gene or protein in a cell. The top 2,000 highly variable genes in the RNA matrix were spliced with the protein matrix and performed a joint normalization as below:

$$x'_{ij} = exp\left(\frac{\sum_j \log(1 + x_{ij})}{\|K_i\|}\right), \quad (3)$$

where x_{ij} represents the expression for gene/protein *i* in cell *j*, and $\|K_j\|$ represents the sum of expressed values in of that gene/protein.

Matched scRNA-seq and scATAC-seq data (scRNA-ATAC-seq)

Data filtering and quality control were performed as usual. We first annotated peak regions in the scATAC-seq based on the method described in MAESTRO[36]. The regulatory potential R_{ij} of each $peak_j$ to each $gene_i$ are calculated independently by the exponential weight decay with the distance from the peak to the transcription start site (TSS):

$$R_{ij} = \begin{cases} 0, & d_{ij} > 150kb \; or \; peak_i \; located \; in \; any \; nearby \; genes \\ \frac{1}{l}, & peak_j \; located \; at \; the \; exons \; region \; of \; the \; gene_i \\ 2^{-\frac{d_{ij}}{d_0}}, & else \end{cases} \quad (4)$$

where d_{ij} is the distance between the center of $peak_j$ and the TSS of $gene_i$. The default

d_0 is set to 10kb. If a $peak_j$ is located at the exon regions of the $gene_i$, the regulatory potential R_{ij} will be normalized by the length of the exon region l. To ensure computational efficiency, we set the R_{ij} as 0 if the $peak_j$ to TSS of $gene_i$ distance is over 150 kb. The default parameters were applied. The gene regulatory activity A_{ik} of $gene_i$ in $cell_k$ is defined as:

$$A_{ik} = \sum_i R_{ij} P_{jk}\,, \tag{5}$$

where P_{jk} is the binary value in the scATAC-seq count matrix of $peak_j$ in $cell_k$.

We assume that the activity of a gene to a cell is determined by both gene expression activity and gene regulatory activity while with different contributions. Different than the contribution weights determined directly based on the expression and chromatin accessibility values in Seurat v4 (weighted nearest neighbor)[5], we hypothesized that the contribution relation of gene expression and chromatin accessibility is dynamic to the future state of the cell, which can be estimated by RNA velocity. The RNA velocity is determined by the abundance of unspliced and spliced mRNA in a cell. The amount of unspliced mRNA is determined by gene regulation and gene transcription rate (which is considered to be the same among genes), and the amount of spliced mRNA is determined by the difference between unsliced mRNA and degraded mRNA. We assume that for genes with positive RNA velocities, there are higher potentials to drive genes to be transcribed. Thus, its regulatory activity related to chromatin accessibility has a higher influence than the gene expression in defining the overall gene activity in a cell of the current snapshot; for genes with negative velocities, on the other hand, the transcription rate tend to be decelerated, and regulatory activity has less influence to the cell than gene expression activity. We define a gene activity score (GAS) of $gene_i$ in $cell_k$, which integrates RNA and ATAC information, is defined as:

$$GAS_{ik} = \begin{cases} E_{ik}.\sum_k E_{ik} + (1+b^{v+})A_{ik}.\left((1+b^{v+})\sum_k A_{ik}\right), & for\ v_{ik} > 0 \\ E_{ik}.\sum_k E_{ik} + (1-b^{v-})A_{ik}.\left((1+b^{v+})\sum_k A_{ik}\right), & for\ v_{ik} < 0 \\ E_{ik}.\sum_k E_{ik} + A_{ik}.\sum_k A_{ik}\,, & for\ v_{ik} = 0 \end{cases} \tag{6}$$

where E_{ik} and A_{ik} represents the gene expression activity and gene regulatory activity, respectively, of $gene_i$ in $cell_k$ which are normalized by rows. v_{ik} represents the RNA velocity of $gene_i$ in $cell_k$, calculated by using CellRank[37]. The weighted of scATAC-seq is defined as:

$$b^+ = \frac{\sqrt{\left(\left|V_{,k}^+\right| - rank\left(v_{ik}\middle|V_{,k}^+\right)\right)^2 + \left(\left|V_{i,}^+\right| - rank\left(v_{ik}\middle|V_{i,}^+\right)\right)^2}}{\sqrt{\left(\left|V_{,k}^+\right| - 1\right)^2 + \left(\left|V_{i,}^+\right| - 1\right)^2)}} \tag{7}$$

$$b^{-} = \frac{\sqrt{\left(\left|V_{,k}^{-}\right| - rank\left(v_{ik}\middle|V_{,k}^{-}\right)\right)^{2} + \left(\left|V_{i,}^{-}\right| - rank\left(v_{ik}\middle|V_{i,}^{-}\right)\right)^{2}}}{\sqrt{\left(\left|V_{,k}^{-}\right| - 1\right)^{2} + \left(\left|V_{i,}^{-}\right| - 1\right)^{2})}} \tag{8}$$

where $V_{i,}^{+}$ is the positive velocity in $gene_i$, $V_{i,}^{-}$ is the negative velocity in $gene_i$. And $V_{,k}^{+}$ is the positive velocity in $cell_k$, $V_{,k}^{-}$ is the negative velocity in $cell_k$.

Construction of gene-cell heterogeneous graph

For any scMulti-omics data type, we now obtain a matrix that integrates information from both modalities, with columns as cells and rows as genes only. Values in the integrated matrix represent for either normalized gene expressions (for multiple scRNA-seq and CITE-seq) or GAS (for scRNA-ATAC-seq). Given an integrated matrix $X \in \mathbb{R}^{NXM}$ with N cells and M genes generated in the last step, we denote a heterogeneous graph $G = (V_C, V_G, E)$, where $V_C = \{c_1, c_2, \dots, c_N\} \in \mathbb{R}^N$ and $V_G = \{g_1, g_1, \dots g_M\} \in \mathbb{R}^M$ represents for all cells and genes in X, respectively, and $E = \{e_{ij}\}$ represents for the edge between c_i and g_j with the weight corresponding to gene expression of g_j in c_i. To learn an initial embedding of each node (c_i and g_j) in the matrix X, a feature autoencoder with two deep learning layers of dense networks in both encoder and decoder was used. The training objective of this feature autoencoder is to achieve a maximum similarity between X and reconstructed matrix X' by minimizing the mean squared error (MSE):

$$\sum \left(X - X_g\right)^2, and \sum (X - X_c)^2 \tag{9}$$

We denote the initial embedding for each c_i and g_j as $H_{c_i}^0$ and $H_{g_j}^0$.

Learning joint embeddings via a heterogeneous graph transformer

We propose an unsupervised HGT framework[14, 15] to learn joint embeddings of c_i and g_j. Given a heterogeneous bipartite graph $G = (V_C, V_G, E)$, DeepMAPS extracts all linked cell-gene node pairs which are denoted as (s, t), where t means the target node and s means the neighbor node of t. The training processes consist of the following four steps: 1) calculate multi-head attention; 2) pass heterogeneous message; 3) aggregate neighbors' information; 4) calculate loss function. We also denote the embedding of the $l\ th$ HGT layer as $\mathcal{H}^l$. To learn joint embeddings of c_i and g_j, we aggregate information from s to get a contextualized representation for t and simultaneously learn the representations of V_C and V_G. It is noteworthy that, to handle the heterogeneous relations in the graph, attention will be calculated via multiple heads, where each node type (cell node and gene node) has a unique head in attention and is linearly projected to a low dimensional space to maximally model the distribution differences.

1) Calculate multi-head attention.

We extract all linked node pairs, where a target node $v_t \in \{V_C, V_G\}$ is directly linked to its

source (neighbor) node $v_s = \{V_C, V_G\}$ through edge e. The contextualized representation of v_t on the l th layer is denoted as $\mathcal{H}^l[v_t]$, which can be learned by its own presentation $\mathcal{H}^{l-1}[v_t]$ and its neighbor $\mathcal{H}^{l-1}[v_s]$ from the $(l-1)^{\text{th}}$ layer, where $l \geq 1$. The overall model is formulated as:

$$\mathcal{H}^l[v_t] \leftarrow \underset{\forall s,t \in V, \forall e \in E}{Aggregate}(\mathcal{H}^{l-1}[Attention(v_s, v_t, e) \cdot Message(v_s)]), \tag{10}$$

where *Attention* estimates the importance of each neighbor; *Message* extracts the information passed from the neighbors; *Aggregate* is the final step to aggregate the neighborhood message by the attention weight.

The multi-head attention is proposed in the attention level to calculate h-head attention for each edge $e = (v_s, v_t)$. Each target node t in the head $h \epsilon \{1,2, \dots, H\}$ was mapped into a target-node vector $T^h(v_t)$ in each HGT layer via linear projection $T^h_{\tau(t)}: R^d \rightarrow\times\ R^{\frac{d}{H}}$, where d is the dimension of initial node feature, H is the number of heads, and $\frac{d}{H}$ is the feature dimension per head. Similarly, each neighbor node s in the head h was mapped into a key vector $K^i(v_s)$ with a linear projection $K^h_{\tau(s)}: R^d \rightarrow\times\ R^{\frac{d}{H}}$. The similarity between the queries and keys is measured (e.g., scaled dot product operator) as attention.

$$T^h(v_t) = T^h_{\tau(t)}\left(\mathcal{H}^{(l-1)}[v_t]\right), \tag{11}$$

$$K^h(v_s) = K^h_{\tau(s)}\left(\mathcal{H}^{(l-1)}[v_s]\right), \tag{12}$$

Then we calculate the multi-head attention value for source node s to target node t by the dot product. To maximize parameter sharing while still maintaining the specific characteristics of different relations, we propose to parameterize weight matrices $W^{ATT}_{\phi(e)}$ of the interaction operators. The $i\ th$ head attention can be defined as:

$$ATThead^h(v_s, v_t, e) = \left(K^h(v_s) W^{ATT}_{\phi(e)} Q^h(v_t)^T\right) \cdot \frac{\mu\langle \tau(v_s), \tau(v_t)\rangle}{\sqrt{d}}, \tag{13}$$

where T is the transposal function, μ is a prior tensor to denote the significance of each edge e, serving as an adaptive scaling to the attention. The attention score in the $h\ th$ head in the $l\ th$ layer is defined as:

$$Att(v_s, v_t, e) = \underset{\forall s \in V(t)}{\text{Softmax}}\left(\underset{H}{||}\, ATThead^h(v_s, v_t, e)\right). \tag{14}$$

2) Pass heterogeneous message.

To alleviate the distribution differences of different types of nodes and edges, we incorporate the types of edges into the message passing. The $h\ th$ head message for each edge (s, t) can be defined as:

$$M^h(v_s, v_t, e) = M^h_{\tau(s)}\left(\mathcal{H}^{l-1}[v_s]\right)W^{MSG}_{\phi(e)} \tag{15}$$

where each source node s in the head i was mapped into a message vector by a linear projection $M^h_{type(s)}: R^d \rightarrow\times\ R^{\frac{d}{H}}$, $W_e^{Msg} \in R^{\frac{d}{H}\times\frac{d}{H}}$ is a distinct edge-based matrix for each edge, e is the edge type of the heterogeneous graph. After multi-head aggregate, the degree of message passing can be defined as:

$$Msg(v_s, v_t, e) = \underset{H}{||} head^h(v_s, v_t) \tag{16}$$

3) Aggregate neighbors' information.

To obtain the represents of each node, we need to aggregate multi-head attention and message. The attention vectors can be regarded as the weight for message representation. The representation of target nodes $\mathcal{H}^l\ [v_t]$ can be updated as:

$$\mathcal{H}^l[v_t] = A_{Linear_{type(t)}}\left(\sigma\left(\bigoplus_{\forall s\in N(t)}\left(Att(v_s, v_t, e)\cdot Msg(v_s, v_t, e)\right)\right)\right) + \mathcal{H}^{l-1}[v_t] \tag{17}$$

4) Calculate loss function.

The original application of HGT was to solve node classification problems. Here, to train cell and gene embeddings without supervised classification labels, we use a graph autoencoder (GAE) framework. The whole HGT structure was deployed as an encoder in the GAE, and we defined two embedding matrices E_c and E_g recording the trained embeddings of cells and genes from the HGT encoder with both columns representing the same embedding dimensions, and rows representing cells and genes, respectively. A decoder was used to reconstruct the heterogeneous graph by the inner product of E_c and E_g. The loss function of GAE is defined as:

$$loss = \sum_i [p(x_i)logp(x_i) - p(x_i)logq(x_i)], \tag{18}$$

where $p(x_i)$ and $q(x_i)$ represents the softmax operation for the $i\ th$ row of E_c and E_g.

HGT training on subgraphs

To handle the efficiency and capability of applying HGT on such a giant heterogeneous graph (tens of thousands of nodes and millions of edges), we performed model training on subgraphs and multiple mini-batches based on the idea of HGSampling[14]. The core of HGSampling is to sample heterogeneous subgraphs with similar proportions in different type of nodes which can avoid sampling highly imbalanced subgraph in training process. We sample n_batch number of subgraphs for each epoch training, and the one-hot of each node will be put into the trained model to obtain all node embedding. Given a node t which has been sampled, a dictionary $D[\tau]$ for each node type τ, we add all the first-neighbor node of *t* into the corresponding $D[\tau]$ and add *t*'s normalized degree to these neighbors to

calculate the sampling probability *p*.

$$p[\tau][s] = \frac{D[\tau][s]^2}{\left||D[\tau]|\right|^2}, \tag{19}$$

where $p[\tau][s]$ is the sampling probability for each source node s of type τ, $D[\tau]$ is all node for type τ with the normalized degree, ||.|| is the 2-norm, $D[\tau][s]$ is the normalizer degree for source node s of node type τ. Then we sampled all types of nodes according to the probability in $D[\tau]$, and moved them out of $D[\tau]$. We repeated this sampling for 50 times to obtain 50 subgraphs that maximize the coverage of the whole heterogeneous graph, and each subgraph was trained with 100 epochs.

Cell clustering and cell-type-active gene association network prediction

Cell clustering.

We applied the Louvain clustering method (igraph v1.2.7, R package) to predict cell clusters on cell-embedding matrix E_c.

Attention-based gene module detection.

To infer the connection of genes and cell clusters, we extract the attention value of gene g to cell c in head i through the step of multi-head attention calculation. We define the importance I of g to c as:

$$I(g,c) = \sqrt{\sum_{H} ATThead^h(g,c)^{\,2}} \tag{20}$$

We assign genes to each cell with a threshold of $mean_g\big(I(g,c)\big) + sd_g(I(g,c))$. The gene g is considered to be one of the active genes in cell c, if the I is higher than the threshold.

The Steiner Forest Problem (SFP) model

We build an SFP model on a heterogeneous graph to extract the most critical gene-gene and gene-cell relations contributing to the gene module specificity in a cell cluster. The input of this model includes three parts:

1. Gene-gene relations defined by the embedding (resulted from GAE) Pearson's correlation between genes (E),
2. Gene-cell relations are defined by the attention score of a gene to a cell (F),
3. A set of cell clusters, $\{V_i, i = 1, 2, \cdots, k\}$, predicted by the HGT model.

We define a weighted heterogeneous graph, $G = (U \cup V, E \cup F)$, in which nodes represent genes (U) and cells ($V = \cup_i V_i$), edges S represent both gene-gene (E) and gene-cell (F) relations. We formulate this problem using a combinatorial optimization model defined as below

$$\min_{S \subseteq E \cup F} \sum_{e \in S} w(e) \tag{21}$$

s.t.

$$r_S(u, v) = 1, \forall\, u, v \in V_i, i = 1, 2, \cdots, k. \tag{22}$$

where $r_S(u, v)$ is a binary indicator function to represent whether two nodes,u and v , are connected (1) or not (0) in the subgraph induced by S in G. We aim to identify the minimum weighted edge set, S, from the heterogeneous graph G, so that cells in the same cell type could be connected to each other via edges in S.

First, in view of the huge size of G, containing $|U \cup V|$ nodes and $|E \cup F|$ edges, we convert G into a sparser graph, G', by iteratively finding a global alignment between genes and cells based on the gene-cell edges, using the maximum matching theory[38]. In graph theory, a matching or independent edge set in an undirected graph is a set of edges without common nodes, and a maximum matching in a weighted graph is a matching M that yields the maximum sum of edge weights. To fulfill this task, we build a weighted bipartite graph, $G^B = (U \cup V, F)$, by only retaining the gene-cell edges, i.e., F. The objective is to identify an edge subset, M, to satisfy

$$\max_{M \subseteq F} \sum_{e \in M} w(e)\,. \tag{23}$$

We calculate the M of G^B using the igraph R package[39]. Then, we remove the cell nodes incident by edges in M. Repeat the prediction of maximum matching and deletion of cell nodes incident to edges in previously identified maximum matchings, until there is no cell node in the remaining graph. We compute the union of all the matchings as F', and then construct $G' = (U \cup V, E \cup F')$. Finally, the weights of gene-cell edges, F', and gene-gene edges, E, are normalized by the following two functions, respectively.

$$\frac{\max\big(w(F')\big) - w(F')}{\max\big(w(F')\big) - \min\big(w(F')\big)} \tag{24}$$

$$\frac{\max\big(w(E)\big) - w(E)}{\max\big(w(E)\big) - \min\big(w(E)\big)} \tag{25}$$

Second, we find the edge set, S, of the Steiner forest, T, as follows. To begin with, we utilize the igraph R package to calculate a minimum spanning forest (MSF)[38], T, of G'. A MSF means that each pair of nodes in the same connected component could be connected to each other. Herein, we only need the edges to connect cell nodes belonging to the same cell type. Therefore, we iteratively remove the gene nodes with degree one from T, until no gene node with degree one exists in T. Finally, we output the edge set of

T, i.e., S, as the solution to the SFP model.

For each connected component of the Steiner forest, the gene-gene edges denote the co-expression relations among genes in the same module, while the set of gene-cell edges represents the cell type specificity of this gene module, and this module is a cell-type-active gene module.

Construct gene regulatory network from scRNA-ATAC-seq data

Infer master TFs and GRNs in each cell type.

To quantify the intensity of genes regulated by TFs, we design regulatory intensive (RI) score, which can be decomposed into two components as: 1) the regulatory potential (R_{ij}) of peaks calculated in the preprocessing step, and 2) the binding affinity (BA) score of TFs to the peaks. The TF binding profiles were obtained from JASPAR database. To reduce false positives of binding site, we select significance binding sites with transformed *p*-values for TF binding profile matches less than 1e-04. The BA score is the transformed relative score which obtained from TF binding profiles. Then the RI score $RI_{L,i,k}$ of TF L to the gene i in the cell k is defined as:

$$RI_{L,i,k} = \sum_{p} BA_{L,p} \cdot R_{ij} \tag{26}$$

Master TFs are genes at the top of a gene regulation hierarchy, particularly in regulatory pathways related to cell fate and differentiation. To infer cell type master TFs, we construct cell-type-specific GRN with RI score as edges weight and calculate centrality which reflects the importance of each node in the network to rank the TFs in each cell type. TFs with high ranked are regarded as master TFs. Consider the RI score of TFs to genes, eigenvector centrality which assigns relative scores to all nodes in the network based on the concept that connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes is applied to infer master TFs. The eigenvector centrality of a node v in GRN can be defined as:

$$C_v = \alpha_{max}(v) \tag{27}$$

Where α_{max} is the eigenvector corresponding to the largest eigenvalue of the weighted adjacency matrix of a GRN.

Identity differential regulon (CTSRs)

To detect regulon associated with disease states, we identity CTSRs by logFC and Wilcoxon rank-sum test. For a cell type active regulon, we define a regulon activity score (RAS) as:

$$RAS(TF, CT) = \frac{\sum_i \sum_{k \in CT} GAS_{i,k} \,.\, RI_{L,i,k}}{NG}, \tag{28}$$

where L means TF, i means gene and k means cell. N is the cell number in a cell type, G is the gene number in the regulon $L - CT$. Then we construct a RAS matrix with $|L - CT|$ rows, $|CT|$ columns. The significance of difference is calculated using the Wilcoxon rank-

sum test. If the BH-adjusted *p*-value is less than 0.05 between different cell clusters and log fold change larger than 0.25, we consider the regulon is differentially active in this cluster and defined as a CTSR.

Benchmarking quantification and statistics

Adjusted rand index (ARI)

ARI is used to compute similarities by considering all pairs of the samples that are assigned in clusters in the current and previous clustering adjusted by random permutation. To calculate ARI, a contingency table is built to summarize the overlaps between the two cell label lists with n elements (cells). Each entry denotes the number of objects in common between the two label lists. The ARI score can be calculated as:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]} \tag{29}$$

where RI is the unadjusted rand index which defined as:

$$RI = \frac{a + b}{C_n^2} \tag{30}$$

Average Silhouette Weight (ASW)

Different from ARI which requires known ground truth labels, silhouette refers to a method of interpretation and validation of consistency within clusters of data. The silhouette value is a measure of how similar an object is to its luster (cohesion) compared to other clusters (separation). The silhouette ranges from −1 to +1, where a high value indicates that the object is well matched to its cluster and poorly matched to neighboring clusters. The silhouette score $s(i)$ can be calculated by:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} = \begin{cases} 1 - \dfrac{a(i)}{b(i)}, & if\ a(i) < b(i) \\ 0, & if\ a(i) = b(i) \\ \dfrac{b(i)}{a(i)} - 1, & if\ a(i) > b(i) \end{cases} \tag{31}$$

where $a(i)$ is the average distance between a cell i and all other cells in the same cluster, and $b(i)$ be the average distance of i to all cell in the nearest cluster to which i does not belong. We take the average silhouette of all cells in a cluster as the average silhouette weight (ASW) to represent the whole cell cluster.

Closeness centrality (CC)

The closeness centrality (CC)[40] of a vertex u is defined by the inverse of the sum length of the shortest paths to all the other vertices v in the undirected weighted graph. The formulation is defined as:

$$CC(u) = \frac{1}{\sum_{i \neq j} d_w(u, v)} \tag{32}$$

Where $d_w(u, v)$ is the shortest weighted paths between u and v. If there is no path

between vertex u and v, the total number of vertices is used in the formula instead of the path length. The CC is calculated by R package igraph with function igraph::betweenness.

Eigenvector centrality (EC)

Eigenvector centrality (EC)[41] scores correspond to the values of the first eigenvector of the graph adjacency matrix. The EC score of vertex u is defined as:

$$EC(u) = x_u = c \sum_{v \in g} a_{uv} x_v \tag{33}$$

Where c is inverse of the eigenvalues of eigenvector $x = [x_1, x_2, \dots, x_n]$, a_{uv} is the weighted adjacent matrix of undirect graph g. The EC is calculated by R package igraph with function igraph::evcent.

Pathway enrichment test

To evaluate the function of the regulatory network, we use pathway enrichment analysis[42] to identify pathways that are significantly represented in each cell cluster active regulon, and count the number of regulon-enriched pathways. The pathway enrichment analysis is done by R package enrichR[43].

Robustness evaluation

Cell cluster leave-out test

For a benchmark dataset with a real cell type label, we removed all cells in one cell type and ran DeepMAPS. We traverse all cell types (one at a time) to evaluate the robustness with ARI. For data without benchmark labels, we removed cells in predicted cell clusters from DeepMAPS and other benchmark tools, respectively.

Read depth simulation test

We performed a downsampling simulation for gene expressions to test the robustness of DeepMAPS to read depth. Let matrix C be the $N \times M$ expression count matrix, where N is the number of cells and M is the number of genes. Define the cell sequencing depths $c_i = \sum_{j=1}^{M} C_{ij}$, i.e., the column sums of C. Thus, the average sequencing depth of the experiment is $\bar{c} = \frac{\sum_{i=1}^{n} c_i}{N}$. Let $t < \bar{c}$ be our target downsampled sequencing depth and let C^* be the $N \times M$ downsampled matrix. We perform the downsampling as follows:

For each spot $i = 1, \dots, N$:

1) Define the total counts to be sampled in the cell i as $t_i = \frac{t \times c_i}{\bar{c}}$.
2) Construct the character vector of genes to be sampled as $G_i = \{\underbrace{1, \dots, 1}_{C_{i1}}, \underbrace{2, \dots, 2}_{C_{i2}}, \dots, \underbrace{M, \dots, M}_{C_{iM}}\}$.
3) Sample t_i elements from G_i without replacement and define N_j as the number of times gene j was sampled from G_i for $j = 1, \dots, M$.

4) Let $C_{ij}^{*} = N_j$.

Using this method, the average downsampled sequencing depth is:

$$\overline{C}^{*} = \frac{\frac{t}{\bar{c}}c_1 + \frac{t}{\bar{c}}c_2 + \cdots + \frac{t}{\bar{c}}c_n}{N} = \frac{\frac{t}{\bar{c}}\sum_{i=1}^{N} c_i}{N} = \frac{t}{\bar{c}} \times \bar{c} = t \tag{34}$$

as desired. Note that, this method preserves the relative total counts of each cell, i.e., cells that had higher sequencing depths in the original matrix have proportionally higher depths in the downsampled matrix.

Comparisons with existing tools

In order to assess the performance of DeepMAPS alongside other proposed scMulti-omics benchmark tools, we compare DeepMAPS with Seurat (v 3.2.3 and v 4.0.0, https://github.com/satijalab/seurat), MOFA+ (v 1.0.0, https://github.com/bioFAM/MOFA2), Harmony (v 0.1, https://github.com/immunogenomics/harmony), and TotalVI (v 0.10.0, https://github.com/YosefLab/scvi-tools). Due to the integration capability, DeepMAPS was compared with Seurat v 3.2.3 and Harmony on multiple scRNA-seq data, with Seurat v4.0.0, MOFA+, and TotalVI on CITE-seq data, and with Seurat 4.0.0 and MOFA+ on scRNA-ATAC-seq data. All benchmark tools used the default settings. We also evaluated the performance of gene association network inference with IRIS3[11] and a normal gene co-expression network inference method. Specifically, in IRIS3, cell-gene biclusters were first identified based on the QUBIC2 algorithm. Cell-type-active gene modules were identified in each cell cluster (using the same cell label predicted in DeepMAPS to ensure the comparability) by performing a cell-wise hypergeometric enrichment. On the other hand, all genes were selected to calculate a gene expression correlation score (Pearson's correlation) between any pairs of two genes using cells in one cell cluster. Gene pair expression correlations with a BH-adjusted p-value smaller than 0.05 were kept and used to build the overall co-expression network in one cell cluster. Co-expressed sub gene modules were inferred by performing Louvain clustering on the co-expression network. For scRNA-ATAC-seq data, we compared regulon and cell-type-specific regulon inferred from DeepMAPS with IRIS3 in terms of enriched biological pathways. Noted that, IRIS3 only supports regulon inference from scRNA-seq data based on de novo motif findings; thus, here, we used the GAS matrix generated in DeepMAPS as an input of IRIS3.

DeepMAPS server construction

DeepMAPS runs on an HPE XL675d RHEL system with 2 x 128-core AMD EPYC 7H12 CPU, 64GB RAM, and 2 x NVIDIA A100 40GB GPU. The backend is written in TypeScript using the NestJs framework. Auth0 is used as an independent module to provide user authentication and authorization services. Redis houses a queue of all pending analysis jobs. There are two types of jobs in DeepMAPS: The stateful jobs are handled by the Plumber R package to provide real-time interactive analysis; The stateless jobs, such as CPU-bound bioinformatics pipelines and GPU training tasks that could take a very long

time, are constructed using Nextflow. All the running jobs are orchestrated using Nomad, allowing each job to be assigned with proper cores and storage, as well as keeping the jobs scalable based on the server load. The job results are deposited to a MySQL database. The frontend is built with NUXT, Vuetify as the UI library, Apache ECharts, and Cytoscape.js for data visualization. The frontend server and backend server are communicated using REST API.

Data availability

All data used for benchmarking and case study are collected from the public domain and can be retrieved using links or accession numbers provided in **Supplementary Tab. 1.**

Code availability

The source code of DeepMAPS Docker is freely available at (https://github.com/OSU-BMBL/deepmaps). The DeepMAPS webserver is available at https://bmblx.bmi.osumc.edu/.

References

1. Stuart, T. & Satija, R. Integrative single-cell analysis. *Nat Rev Genet* **20**, 257-272 (2019).
2. Ma, A., McDermaid, A., Xu, J., Chang, Y. & Ma, Q. Integrative Methods and Practical Challenges for Single-Cell Multi-omics. *Trends in Biotechnology* (2020).
3. Argelaguet, R., Cuomo, A.S.E., Stegle, O. & Marioni, J.C. Computational principles and challenges in single-cell data integration. *Nat Biotechnol* (2021).
4. S Teichmann, M.E. Method of the Year 2019: Single-cell multimodal omics. *Nat Methods* **17**, 1 (2020).
5. Hao, Y. et al. Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573-3587.e3529 (2021).
6. Argelaguet, R. et al. MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. *Genome Biology* **21**, 111 (2020).
7. Korsunsky, I. et al. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nature Methods* **16**, 1289-1296 (2019).
8. Gayoso, A. et al. Joint probabilistic modeling of single-cell multi-omic data with totalVI. *Nature Methods* **18**, 272-282 (2021).
9. Li, Y. et al. Elucidation of Biological Networks across Complex Diseases Using Single-Cell Omics. *Trends Genet* (2020).
10. Aibar, S. et al. SCENIC: single-cell regulatory network inference and clustering. *Nat Methods* **14**, 1083-1086 (2017).
11. Ma, A. et al. IRIS3: integrated cell-type-specific regulon inference server from single-cell RNA-Seq. *Nucleic Acids Res* **48**, W275-W286 (2020).
12. Han, P., Gopalakrishnan, C., Yu, H. & Wang, E. Gene Regulatory Network Rewiring in the Immune Cells Associated with Cancer. *Genes (Basel)* **8**, 308 (2017).
13. Wang, J. et al. scGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. *Nature Communications* **12**, 1882 (2021).
14. Hu, Z., Dong, Y., Wang, K. & Sun, Y. in Proceedings of The Web Conference 2020 2704–2710 (Association for Computing Machinery, Taipei, Taiwan; 2020).
15. Wang, X. et al. in The World Wide Web Conference 2022–2032 (Association for Computing Machinery, San Francisco, CA, USA; 2019).
16. Zou, D. et al. in Advances in Neural Information Processing Systems 11247-11256 (2019).
17. Jin, S. et al. Inference and analysis of cell-cell communication using CellChat. *Nature Communications* **12**, 1088 (2021).
18. Ampudia, J. et al. CD6-ALCAM signaling regulates multiple effector/memory T cell functions. *The Journal of Immunology* **204**, 150.113-150.113 (2020).
19. Skonier, J.E. et al. Mutational analysis of the CD6 ligand binding domain. *Protein Engineering, Design and Selection* **10**, 943-947 (1997).
20. Gimferrer, I. et al. Relevance of CD6-mediated interactions in T cell activation and proliferation. *Journal of immunology (Baltimore, Md. : 1950)* **173**, 2262-2270

(2004).
21. Johnston, R.J., Lee, P.S., Strop, P. & Smyth, M.J. Cancer Immunotherapy and the Nectin Family. *Annual Review of Cancer Biology* **5**, 203-219 (2021).
22. Li, X.-Y. et al. CD155 loss enhances tumor suppression via combined host and tumor-intrinsic mechanisms. *J Clin Invest* **128**, 2613-2625 (2018).
23. Li, K. et al. cellxgene VIP unleashes full power of interactive visualization, plotting and analysis of scRNA-seq data in the scale of millions of cells. *bioRxiv*, 2020.2008.2028.270652 (2020).
24. Pereira, W. et al. Asc-Seurat – Analytical single-cell Seurat-based web application. *bioRxiv*, 2021.2003.2019.436196 (2021).
25. Gardeux, V., David, F.P.A., Shajkofci, A., Schwalie, P.C. & Deplancke, B. ASAP: a web-based platform for the analysis and interactive visualization of single-cell RNA-seq data. *Bioinformatics* **33**, 3123-3125 (2017).
26. Li, B. et al. Cumulus provides cloud-based data analysis for large-scale single-cell and single-nucleus RNA-seq. *Nature Methods* **17**, 793-798 (2020).
27. Hillje, R., Pelicci, P.G. & Luzi, L. Cerebro: interactive visualization of scRNA-seq data. *Bioinformatics* **36**, 2311-2313 (2019).
28. Prompsy, P. et al. Interactive analysis of single-cell epigenomic landscapes with ChromSCape. *Nature Communications* **11**, 5702 (2020).
29. Bolisetty, M.T., Stitzel, M.L. & Robson, P. CellView: Interactive exploration of high dimensional single cell RNA-seq data. *bioRxiv*, 123810 (2017).
30. Mohanraj, S. et al. CReSCENT: CanceR Single Cell ExpressioN Toolkit. *Nucleic Acids Research* **48**, W372-W379 (2020).
31. Patel, M.V. iS-CellR: a user-friendly tool for analyzing and visualizing single-cell RNA sequencing data. *Bioinformatics* **34**, 4305-4306 (2018).
32. Yousif, A., Drou, N., Rowe, J., Khalfan, M. & Gunsalus, K.C. NASQAR: a web-based platform for high-throughput sequencing data analysis and visualization. *BMC Bioinformatics* **21**, 267 (2020).
33. Zhu, Q. et al. PIVOT: platform for interactive analysis and visualization of transcriptomics data. *BMC Bioinformatics* **19**, 6 (2018).
34. Innes, B. & Bader, G. scClustViz Single-cell RNAseq cluster assessment and visualization [version 2; peer review: 2 approved]. *F1000Research* **7** (2019).
35. Granja, J.M. et al. ArchR is a scalable software package for integrative single-cell chromatin accessibility analysis. *Nat Genet* **53**, 403-411 (2021).
36. Wang, C. et al. Integrative analyses of single-cell transcriptome and regulome using MAESTRO. *Genome Biology* **21**, 198 (2020).
37. Lange, M. et al. CellRank for directed single-cell fate mapping. *bioRxiv*, 2020.2010.2019.345983 (2020).
38. Diestel, R., Schrijver, A. & Seymour, P.D. in MATHEMATISCHES FORSCHUNGSINSTITUT OBERWOLFACH REPORT NO. 16/2007 (Citeseer, 2007).

39. Csardi, G. & Nepusz, T. The igraph software package for complex network research. *InterJournal, complex systems* **1695**, 1-9 (2006).
40. Li, G.S., Li, M., Wang, J.X., Li, Y.H. & Pan, Y. United Neighborhood Closeness Centrality and Orthology for Predicting Essential Proteins. *Ieee Acm T Comput Bi* **17**, 1451-1458 (2020).
41. Parisutham, N. & Rethnasamy, N. Eigenvector centrality based algorithm for finding a maximal common connected vertex induced molecular substructure of two chemical graphs. *J Mol Struct* **1244** (2021).
42. Reimand, J. et al. Pathway enrichment analysis and visualization of omics data using g:Profiler, GSEA, Cytoscape and EnrichmentMap. *Nat Protoc* **14**, 482-517 (2019).
43. Kuleshov, M.V. et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Res* **44**, W90-97 (2016).

Acknowledgments

This work was supported by awards R01-GM131399, R35-GM126985, and U54AG075931 from the National Institute of General Medical Sciences of the National Institutes of Health. The work was also supported by award NSF1945971 from the National Science Foundation. In addition, we thank Dr. Xin Gang from the Ohio State University (USA) for his contribution in helping annotate cell types in the three case studies, and Dr. Fei He from the Northeast Normal University (China) for his valued suggestions in framework construction and data testing.

Contributions

Q.M. and D.X. conceived the basic idea and designed the framework. X.W. and B.L. wrote the DeepMAPS code. C.W. built the backend and frontend server. S.G. designed dynamic figures on the server. Y.Liu carried out RNA velocity calculation. Y.Li designed the SFP model for gene module prediction. A.M, X.W., and J.L. carried out benchmark experiments. X.W., Y.C., and B.L. performed robustness tests. A.M., J.L., and T.X. carried out the case study. J.W., D.W., Y.J., J.L., and L.S. performed tool optimizations. A.M. lead the figure design and manuscript writing. All authors participated in interpretation and writing the manuscript.

Corresponding author

Correspondence to Prof. Qin Ma: Email: qin.ma@osumc.edu; Prof. Dong Xu: xudong@missouri.edu; Prof. Bingqiang Liu: bingqiang@sdu.edu.cn.

Ethics declarations

Competing interests

The authors declare no competing interests.