Article Type: Original Research

Title: Generative pretraining from large-scale transcriptomes: Implications for single-cell deciphering and clinical translation

Hongru Shen¹, Xilin Shen¹, Jiani Hu¹, Jilei Liu¹, Chao Zhang², Dan Wu¹, Mengyao Feng¹, Meng Yang¹, Yang Li¹, Yichen Yang¹, Wei Wang³, Qiang Zhang⁴, Jilong Yang², Kexin Chen^{3#}, Xiangchun Li^{1#}

Affiliations

¹Tianjin Cancer Institute, National Clinical Research Center for Cancer, Key Laboratory of Cancer Prevention and Therapy, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin, China.

²Department of Bone and Soft Tissue Tumor, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin, China.

³Department of Epidemiology and Biostatistics, Key Laboratory of Molecular Cancer Epidemiology of Tianjin, National Clinical Research Center for Cancer, Key Laboratory of Cancer Prevention and Therapy, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin, China.

⁴Department of Maxillofacial and Otorhinolaryngology Oncology, Tianjin Medical University Cancer Institute and Hospital, Tianjin Medical University, Tianjin, China.

Running title: Generative pretraining for large-scale transcriptomes.

Correspondence to:

Prof. Xiangchun Li, Tianjin Cancer Institute, Tianjin Medical University Cancer Institute and Hospital, Huanhu Xi Road, Tiyan Bei, Hexi District, Tianjin, 300060, China. Tel (Fax): (86) 022 - 23372231; Email: lixiangchun2014@foxmail.com

Prof. Kexin Chen, Department of Epidemiology and Biostatistics, Tianjin Medical University Cancer Institute and Hospital, Huanhu Xi Road, Tiyan Bei, Hexi District, Tianjin, 300060, China. Tel: (86) 022 - 2237 2231; Email: chenkexin@tmu.edu.cn

Word counts: 3848

Abstract

Exponential accumulation of single-cell transcriptomes poses great challenge for efficient assimilation. Here, we present an approach entitled *tGPT* towards integration of 22.3 million single-cell transcriptomes by modeling gene expression rankings as generative pretraining task. *tGPT* is conceptually simple in that it autoregressively models the ranking of a gene in the context of its preceding neighbors. We demonstrated the high performance of *tGPT* on a range of fundamental single-cell analysis tasks and novel applications on bulk tissues. The single-cell clusters and cell lineage trajectories derived from *tGPT* are highly aligned with known cell labels and states. The feature patterns of tumor bulk tissues learned by *tGPT* are associated with a wide range of genomic alteration events, prognosis and treatment outcome of immunotherapy. *tGPT* represents a new analytical paradigm for integrating and deciphering massive amount of transcriptome data and it will facilitate the interpretation and clinical translation of single-cell transcriptomes.

Key words: single-cell, transcriptome, generative pretraining, deep learning

Introduction

Rapid advancement in single-cell RNA sequencing leads to dramatical drop in sequencing cost and allows for millions of single-cell transcriptomes to be digitized in a single experiment simultaneously. The whole human body is estimated to have 30 trillion cells. Single-cell transcriptome sequencing provided an unprecedented resolution to distinguish different cell type clusters, depict hierarchical cell arrangement and decipher transitional cell states. To achieve this goal, multiple single-cell atlasing projects have been established internationally, including Human Cell Atlas (HCA)¹, Single Cell Expression Atlas (SCEA)², COVID-19 Atlas³, Tabula Muris Atlas⁴ and Mouse Cell Atlas⁵. The HCA project¹ aims to digitize all cells and create a reference map of the human body through community-driven initiative that researchers all around the world can contribute. SCEA² compiles and annotates publicly available single-cell transcriptomes across multiple species and different studies. The COVID-19 Atlas³ aims at elucidating molecular mechanism and therapeutic target of COVID-19 by generating single-cell atlas of SARS-CoV-2 infection in COVID-19 patients. The Tabula Muris⁴ and MCA⁵ atlases constitute the single-cell reference maps of mouse with millions of cells obtained from different organs. These atlasing projects pose tremendous challenge in the integration of diverse transcriptomes from different projects. However, single-cell transcriptomes are generated by different platforms and experimental protocols. They are sparse, noise and prone to batch-effect^{6,7}. Therefore, an analytical method to efficiently integrate ten millions of cells are urgently needed.

Over the past few years, deep learning approaches have led to seismic changes in image recognition and natural language understanding. The success of deep learning could largely attribute to the availability of big data, advancement in computational infrastructure, expressivity and scalability of the computational model. The deep learning model could adeptly

handle super large-scale high dimensional data and assimilate real-world information. Due to the exponential accumulation of millions of cell transcriptomes, elucidation of the reference map of single-cell transcriptomes with deep learning becomes an attractive application. Deep learning methods such as *scVI*⁸, *SAUCIE*⁹ and *INSCT*¹⁰ have been developed for the analysis of single-cell transcriptomes.

The progress of artificial intelligence is undergoing a paradigm shift in computer vision and natural language processing. Deep neural networks based on transformer are becoming the de facto approach in wide variety of scenarios such as vision, language and reasoning¹¹. Transformer-based models pretrained on broad data at scale continues to achieve state-of-the-art progress in image classification^{12,13} and language understanding¹⁴⁻¹⁶. The success of these pretrained models can be attributed to their high expressivity and scalability enabled by transformer to assimilate feature representation from massive amount of unlabeled data. However, the investigation of single-cell transcriptome pretraining at scale has not been well studied.

In this study, we present a deep learning approach entitled *tGPT* towards integration of unlimited number of cells. *tGPT* is built upon transformer that has been widely used in natural language understanding and image recognition. The transformer is an essential component and key success of foundation models because of its high expressivity and scalability¹¹. *tGPT* takes as input the expression rankings of top-expressing genes rather than the actual expression levels. *tGPT* is conceptually simple and empirically efficient. It models the occurrence of a gene in the context of its preceding neighbors' rankings. We developed *tGPT* with 22.3 million cells and systematically evaluated *tGPT* on several heterogeneous datasets for sensitivity to batch-effect, delineation of clustering performance and inference of developmental lineages. We

applied *tGPT* to bulk cancer tissue sequencing samples and found that features obtained from *tGPT* are significantly associated with diverse genomic alteration events, patients' prognosis and treatment outcome of immunotherapy. *tGPT* represents a new analytical paradigm to integrate and decipher large-scale single-cell transcriptomes. It will facilitate the integration and clinical translation of large volume of single-cell transcriptome data.

Results

An overview of *tGPT* and its downstream applications

The analytical framework of *tGPT* (**Figure 1**) consists of three components: development of *tGPT*, applications of *tGPT* for single-cell clustering and inference of developmental lineage and interrogation of feature representation of bulk tissues in relation to genomic alterations, prognosis and treatment response of immunotherapy.

tGPT is formatted as an autoregressive language model in that the output from the previous step is used as input to the next step. The input to *tGPT* is a sequence of gene symbols that are ranked by their expression levels. The purpose is to predict the index of the next gene in the dictionary in the context of all previous genes. The dictionary consists of 20706 protein-coding genes. *tGPT* is trained as an unsupervised generative pretraining task¹⁶. Specifically, for a given cell, let $\mathcal{G} = \{G_1, G_2, \dots, G_n\}$ denote the gene symbols that are sorted in a descending order according to their expression levels. We use the standard language modeling objective $\mathcal{L}(\mathcal{G}) = \sum_i \log P(G_i | G_{i-k}, \dots, G_{i-1}; \theta)$ to maximize the likelihood. Here, k is the width of context window and θ are the parameters of *tGPT* that is used to model the conditional probability. The neural network consists of 8 transformer decoder blocks¹⁷ with 1024 hidden units and 16 attention heads.

Quantitative evaluations of *tGPT* on clustering

We systematically evaluated the clustering performance of *tGPT* on four heterogeneous single-cell datasets of different sizes (50-586k cells) from different species and two bulk tissue sequencing datasets. These four single-cell datasets include Human Cell Atlas Census of Immune Cells¹⁸ (*HCA*, $n = 282,558$), Human Cell Landscape¹⁹ (*HCL*, $n = 586,135$), *Tabula Muris*⁴ ($n = 54,862$) and *Macaque Retina*²⁰ ($n = 124,965$) dataset (See methods for description). The two bulk tissue datasets are *Genotype-Tissue Expression*²¹ (*GTEX*, $n = 11,688$) derived from 30 organs and *The Cancer Genome Atlas*²² (*TCGA*, $n = 9,318$) consisted of 33 cancer types.

The clustering performance of *tGPT* is robust with respect to the numbers of top-expression genes being used. We found that the performance of *tGPT* pretrained on the ranking of top 62 and 126 genes were comparable across these six datasets (**Supplementary Figure 1**). In addition, we observed that clustering performance on features extracted from different transformer layers [Layer-1, ..., Layer-8] are comparable and better than features extracted from the embedding layer across all these six datasets (**Supplementary Figure 1**). We performed grid-search to identify optimal values of two parameters that are most relevant to clustering (see **Methods**) and reported the best performance for each method. The result from grid-search were provided in **Supplementary Figures 2-7**. Quantitatively, *tGPT* achieved an *NMI* ranged from 0.75 on *HCA* to 0.90 on *GTEX*, *ARI* from 0.53 on *HCL* to 0.84 on *Tabula Muris* and *FMI* from 0.55 on *HCL* to 0.85 on *Tabula Muris* (**Figure 2A**). The clustering performance achieved by *tGPT* are comparable to the other methods such as Scanpy²³, Pegasus²⁴ and scVI⁸ (**Supplementary Figures 8-11**). Grid-search results of these methods were provided in **Supplementary Figure 12**.

Across these datasets, *tGPT* was capable of grouping cells with the same or similar types (**Figure 2B-G**). On the *HCA* dataset, *tGPT* was able to identify cells at different developmental phases. For example, it can delineate B cells of different types such as naïve B cells, precursor B (pre-B) cells and progenitor B (pro-B) cells and homologous cells such as conventional DCs (cDCs) and plasmacytoid DCs (pDC), CD14⁺ and CD16⁺ monocytes. Less represented cell types such as megakaryocytes (0.32%) and MSCs (0.10%) were also captured by *tGPT* (**Figure 2B**). On the *HCL* dataset, *tGPT* was able to distinguish between immune cells and nonimmune cells as well as different cell types from fetus and adult such as fetal enterocytes and adult enterocytes (**Figure 2C and Supplementary Figure 9**). On the *Tabula Muris* dataset, *tGPT* was also able to delineate 55 distinct cell types originated from 20 mouse organs (**Figure 2D and Supplementary Figure 10**). On the *Macaque Retina* dataset, distinctive cell clusters from foveal and peripheral regions of fascicularis retina defined by *tGPT* are well matched with cell types defined in the original literature²⁰ (**Figure 2E**). On the *GTEX* dataset, *tGPT* is able to identify different tissues originated from lineage of organs ($NMI = 0.90$), and samples with similar histological structure are close together such as colon, small intestine and stomach (**Figure 2F**). On the *TCGA* dataset, different cancer types are well separated ($NMI = 0.77$). Cancer types with the same tissue of origin tend to clump together in the feature representation spaces captured by *tGPT*. For example, adenocarcinomas and squamous cell carcinomas are closely related in the UMAP plots, respectively.

We observed that *tGPT* is insensitive to batch effect as benchmarked against with the other methods that support batch-correction such as *ComBat*²⁵, *MNN*²⁶, *Harmony*²⁷, *Seurat*^{28,29}, *BBKNN*³⁰, *Scanorama*³¹, *Pegasus*²⁴, *scVI*⁸, *scArches*³², *iMAP*³³ and *DESC*³⁴ as measured on the *HCA* dataset. *tGPT* achieved a comparable *kBET* acceptance rate³⁵ of 0.87 among the aforementioned batch-correction methods (**Supplementary Figure 13L**). The UMAP plots of

these batch-correction methods and their clustering metrics and grid-search results are provided in **Supplementary Figures 13A-K, 14 and 15**, respectively.

Distinct features learned by *tGPT* are connected to cell types

We observed that the head entropy and importance of different cell types from the *HCA* dataset (See **Methods**) are distinctive from each other. Cells of similar lineages or functions such as T-lineage cells exhibited similar entropy patterns (**Figure 3A**). The head importance is varying considerably for different cell types, however, cells of similar types are alike as compared with the other cell types (**Figure 3B**). For each cell type, we calculated the contribution of each gene on the cell final feature representation (See **Methods**). Cell type specific genes have higher attribution scores (**Figure 3C**). For example, *NKG7*, *FGFBP2*, *PRF1*, *GNLY*, *GZMA* and *GZMB* are highly represented in cytotoxic T cells and NK cells (**Figure 3D**). *PPBP* and *PF4* are also highly represented in megakaryocytes (**Figure 3E**). B-lineage cells have high attribution scores for both *CD79A* and *CD79B*. Attribution scores of *MS4A1* and *MZB1* are relative higher in memory B cells and plasma cells, respectively (**Figure 3F**). The attribution score of *CST3* is higher among CD14⁺ monocytes, CD16⁺ monocytes, cDCs and pDCs. In addition, each specific cell types can be defined by specific genes with high attribution scores, for instance plasmacytoid dendritic cells (pDCs, *IRF7*), conventional dendritic cells (cDC, *FCER1A*), CD14⁺ monocytes (*CD14*) and CD14⁺ monocytes (*FCGR3A*) (**Figure 3G**).

Inference of developmental lineage

We used the feature representations learned by *tGPT* to construct cell pseudo-temporal trajectories on *HCA* and *HCL* datasets (See **Methods**). On the *HCA* dataset, the developmental trajectories originated from stem cells and differentiated towards multiple biologically functional cell branches (**Figure 4A**): HSCs to erythroid cells³⁶ or DCs and monocytes (**Figure**

4B); naïve T cells to cytotoxic T cells and NK cells³⁷ (**Figure 4C**); pro-B cells to pre-B cells, then followed by matured naïve B cells, and finally bifurcated into memory B cells or plasma cells³⁸ (**Figure 4D**). In addition, we observed that the cell state signatures are aligned with cell developmental lineages (See **Methods**). For instance, HSCs and pro-B cells are manifested by apparent progenitor signaling (**Figure 4E**). Naïve and mature T cells are featured by distinguishable patterns (**Figure 4F and G**).

On the *HCL* dataset, the developmental tree depicted three differential trajectories of fetal mesenchymal progenitor cells into different mature cell types (**Figure 4H**) with fetal cells at the center of the landscape. The fetal mesenchymal progenitor cells are differentiated into biologically functional fibroblasts (**Figure 4I**), enterocytes (**Figure 4J**), astrocytes and oligodendrocytes (**Figure 4K**).

Clinical significance of *tGPT* in bulk sequencing sample

Here, we demonstrated that *tGPT* is able to capture clinically significant patterns. On the *TCGA* dataset, we found that the importance scores are varying considerably for different attention heads among different layers. The importance score patterns can cluster different cancer types into distinct groups in that cancer of the same tissue-of-origin are closely related whereas cancers of different origins are well separated (See **Methods**, **Figure 5A**). For example, skin cutaneous melanoma (SKCM) and uveal melanoma (UVM), glioblastoma multiforme (GBM) and brain lower grade glioma (LGG) are respectively located in the same clustering branches. In addition, we examined the association between attention head entropy and molecular alteration events (See **Methods**). There are several attention heads exhibited significant association with tumor mutation burden (TMB) in the TCGA pan-cancer cohort and specifically in bladder urothelial carcinoma (BLCA), LUAD and LUSC (**Figure 5B**). We

observed that attention heads also showed significant association with *TP53* mutations at the pan-cancer level and across 9 cancer types (**Figure 5C**). There are also attention heads exhibited significant association with homologous recombination deficiency (HRD) and genome doubling (**Figure 5D and E**) at the pan-cancer level. The association of attention heads with HRD and genome doubling are statistically significant across 4 and 14 cancer types, respectively. Meanwhile, the attention heads exhibited prognostic significance at pan-cancer level (**Figure 5E**) and across 7 cancer types (**Supplementary Figure 16**).

In addition, we examined the attention head patterns in relation to immunotherapy in an immune checkpoint block (ICB) clinical trial of urothelial carcinoma consisted of 298 patients: 25 patients with CR, 43 with PR, 63 with SD and 63 with PD (See **Methods**). We found that importance and entropy scores are distinguishable amongst patients with different therapeutic outcome (**Figure 5G and H**). We observed gradually varying entropy values from SD to PR to CR by taking the PD baseline (**Figure 5I**) and significant difference among 5 attention heads in patients with CR/PR versus SD/PD (**Figure 5J**). We quantified expression signatures such as tumor evasion and T cell immune infiltration attended by different attention heads (See **Methods**). By taking PD as baseline, we observed a gradually decreasing patterns of tumor evasion and increasing patterns of T cell immune infiltration from SD to PR to CR (**Figure 5K and L**). The attention heads also exhibited prognostic significance in this clinical trial (**Figure 5M and N**).

Discussion

Efficient integration of increasingly large-scale single-cell transcriptomes is urgently needed. Here, we introduced a conceptually simple approach towards the integration of unlimited number of single-cell transcriptomes and its potential clinical translational relevance. The

paradigm underpinning *tGPT* in essence is to predict the occurrence of a given gene with its previous context. We developed *tGPT* on a super large-scale single-cell transcriptomes which consists of 22.3 million cells and systematically evaluated its representation learning ability on different single-cell analysis tasks. We note that *tGPT* was insensitive to batch effect and achieved competitive accuracies as compared with benchmark tools. The purpose of this study is to verify the validity of this new paradigm in deciphering large-scale transcriptome data, especially at the level of single-cell atlas. In addition, we showed that the pretrained *tGPT* model can be applied to bulk tissue sequencing samples to extract a variety of features exhibiting significant association with genomic alteration and potential clinical application.

Artificial intelligence is undergoing a paradigm shift and the pretraining models based on transformer are becoming de facto standard in natural language processing and computer vision, achieving state-of-the-art across a wide range of tasks such as natural language understanding, image classification, video and audio recognition¹¹. Representative pretraining models include BERT¹⁴ and GPT¹⁵. The advantage of these pretraining models lie in its ability to assimilate real-world information from super large-scale unlabeled and high-dimensional data. This advantage brings an attractive solution for deciphering single-cell transcriptomes as millions of cells have been sequenced, which exemplified by 22.3 million cells collected in our study. This number is expected to increase exponentially in years ahead. There is no analytical tool that are designed and evaluated on such large volume of data. The high expressivity and scalability of transformer enable *tGPT* to learn rich representation from transcriptomes in a self-supervised manner. The high clustering performance in single-cell cluster delineation is probably attributable to better feature representation learned by *tGPT*. In addition, feature representation from *tGPT* is insensitive to batch effect as the acceptance rate of *kBET* derived from *tGPT* is evenly distributed among the other tools that explicitly used batch information

for batch-correction. This is probably due to the use of rankings of top expressing genes rather than actual expression levels by *tGPT*. *tGPT* is quite different from the other integration tools^{28,29,23} as the others use the actual expression levels of highly variable genes (HVGs) and the batch information. The independence of *tGPT* on batch information makes it attractive for super large-scale transcriptome integration since the batch information is not always available and often neglected by researchers.

The clustering performance in delineating single-cell clusters is robust with respect to the number of top expressing genes used and feature representation extracted from different *tGPT* transformer layers. The clustering metrics obtained from 62 top-expressing genes are comparable to the use of 126 top-expressing genes (**Supplementary Figure 1**). This suggested that the rankings of 62 top-expressing genes are sufficient for cell cluster definition. The idea underpinning *tGPT* is to predict the occurrence of a gene in the context of the occurrences of its preceding neighbors. This type of pretraining is not directly related to cell clustering. This does not guarantee that feature representation learned by the last transformer layer could give rise to better clustering as compared with representation learned by all its previous layers. In our evaluation, the cluster metrics obtained from different transformer layers are comparable and consistently better than the embedding layer (**Supplementary Figure 1**). In addition, we observed that cell-type specific genes have high attribution scores albeit only the rankings are used during pretraining. This finding to some extent can explain why features derived from *tGPT* could lead to high performance in cell clustering.

A new finding emerged from our study is that the pretrained *tGPT* model can be applied to bulk tissues. On the *GTEX* dataset, the feature representations of different organs extracted from *tGPT* can divide samples into distinct clusters, aligning with organs. On the *TCGA* dataset, we

observed that different cancer types are well separated and cancers of the same origins are more closely related, which is consistent with previous report³⁹. Additionally, the feature patterns of TCGA samples exhibited consistent and significant association with genomic alterations. This indicated that rankings of top-expressing genes carry information about alterations in tumor tissues. Meanwhile, the feature patterns derived from *tGPT* are distinctive among patients with different treatment outcomes for immunotherapy. Taken together, our finding would facilitate translational research enabled by super large-scale transcriptomes.

We focused two main directions of *tGPT* for future development. Firstly, *tGPT* can be used to generate large-scale reference mapping with the availability of large-scale disease reference datasets and phenotypes. Secondly, *tGPT* can be further investigated for clinical application such as treatment guiding and prognostic prediction.

In summary, we systematically verified a new, simple and effective analytical paradigm for super large-scale transcriptome analysis and its implications in clinical translation.

ACKNOWLEDGEMENTS

We are grateful for researchers for their generosity to made their data publicly available. This work was supported by the National Natural Science Foundation of China (Grant No. 31801117 to X.L., 31900471 to M.Y. and 82073287 to Q.Z.), National Key Research and Development Program of China (Grant No. 2021YFC2500400 to K.C.), Program for Changjiang Scholars and Innovative Research Team in University in China (Grant No. IRT_14R40 to K.C.) and Tianjin Municipal Health Commission Foundation (Grant No. RC20027 to Y.L).

AUTHOR CONTRIBUTIONS

Xiangchun Li and Kexin Chen designed and supervised the study; Xiangchun Li and Hongru Shen performed data analysis and wrote the manuscript; Xiangchun Li developed the model; Xiangchun Li, Hongru Shen, Xilin Shen, Chao Zhang, Dan Wu, Mengyao Feng, Jiani Hu, Jilei Liu, Yichen Yang, Yang Li, Meng Yang, Wei Wang and Qiang Zhang collected data; Hongru Shen, Xiangchun Li, Kexin Chen and Jilong Yang revised the manuscript.

DECLARATION OF INTERESTS

The authors declare that they have no conflict of interest.

Data and code availability

All the gene expression matrices were downloaded from public databases. The source list of these datasets was provided in **Supplementary Table 1**. We will release the pretrained *tGPT* model and its training code at Github soon.

References

- 1 Aviv, R. *et al.* The human cell atlas. *Elife* **6** (2017).
- 2 Papatheodorou, I. *et al.* Expression Atlas update: from tissues to single cells. *Nucleic acids research* **48**, D77-D83 (2020).
- 3 Wilk, A. J. *et al.* A single-cell atlas of the peripheral immune response in patients with severe COVID-19. *Nature medicine* **26**, 1070-1076 (2020).
- 4 Tabula Muris, C. *et al.* Single-cell transcriptomics of 20 mouse organs creates a Tabula

- 347 Muris. *Nature* **562**, 367-372, doi:10.1038/s41586-018-0590-4 (2018).
- 348 5 Han, X. *et al.* Mapping the Mouse Cell Atlas by Microwell-Seq. *Cell* **173**, 1307,
349 doi:10.1016/j.cell.2018.05.012 (2018).
- 350 6 Tung, P. Y. *et al.* Batch effects and the effective design of single-cell gene expression
351 studies. *Sci Rep* **7**, 39921, doi:10.1038/srep39921 (2017).
- 352 7 Hicks, S. C., Townes, F. W., Teng, M. & Irizarry, R. A. Missing data and technical
353 variability in single-cell RNA-sequencing experiments. *Biostatistics* **19**, 562-578,
354 doi:10.1093/biostatistics/kxx053 (2018).
- 355 8 Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling
356 for single-cell transcriptomics. *Nat Methods* **15**, 1053-1058, doi:10.1038/s41592-018-
357 0229-2 (2018).
- 358 9 Amodio, M. *et al.* Exploring single-cell data with deep multitasking neural networks.
359 *Nat Methods* **16**, 1139-1145, doi:10.1038/s41592-019-0576-7 (2019).
- 360 10 Simon, L. M., Wang, Y.-Y. & Zhao, Z. Integration of millions of transcriptomes using
361 batch-aware triplet neural networks. *Nature Machine Intelligence*, 1-11 (2021).
- 362 11 Bommasani, R. *et al.* On the opportunities and risks of foundation models. *arXiv*
363 *preprint arXiv:2108.07258* (2021).
- 364 12 Chen, M. *et al.* in *International Conference on Machine Learning*. 1691-1703 (PMLR).
- 365 13 Bao, H., Dong, L. & Wei, F. BEiT: BERT Pre-Training of Image Transformers. *arXiv*
366 *preprint arXiv:2106.08254* (2021).
- 367 14 Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: Pre-training of deep
368 bidirectional transformers for language understanding. *arXiv preprint*
369 *arXiv:1810.04805* (2018).
- 370 15 Radford, A., Narasimhan, K., Salimans, T. & Sutskever, I. Improving language
371 understanding by generative pre-training. (2018).

- 16 Brown, T. B. *et al.* Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020).
- 17 Liu, P. J. *et al.* Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198* (2018).
- 18 Michal Slyper, J. W., Marcin Tabaka, Timothy Tickle, Aviv Regev, Bo Li, Orit Rozenblatt-Rosen, Monika S Kowalczyk, Karthik Shekhar, Orr Ashenberg, Danielle Dionne, Jane Lee. Census of Immune Cells.
- 19 Han, X. *et al.* Construction of a human cell landscape at single-cell level. *Nature* **581**, 303-309, doi:10.1038/s41586-020-2157-4 (2020).
- 20 Peng, Y. R. *et al.* Molecular Classification and Comparative Taxonomics of Foveal and Peripheral Cells in Primate Retina. *Cell* **176**, 1222-1237 e1222, doi:10.1016/j.cell.2019.01.004 (2019).
- 21 Consortium, G. T. Erratum: Genetic effects on gene expression across human tissues. *Nature* **553**, 530, doi:10.1038/nature25160 (2018).
- 22 Thorsson, V. *et al.* The Immune Landscape of Cancer. *Immunity* **48**, 812-830 e814, doi:10.1016/j.immuni.2018.03.023 (2018).
- 23 Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* **19**, 15, doi:10.1186/s13059-017-1382-0 (2018).
- 24 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).
- 25 Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118-127, doi:10.1093/biostatistics/kxj037 (2007).
- 26 Haghverdi, L., Lun, A. T., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nature*

- biotechnology **36**, 421-427 (2018).
- 27 Korsunsky, I. *et al.* Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat Methods* **16**, 1289-1296, doi:10.1038/s41592-019-0619-0 (2019).
- 28 Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* **36**, 411-420, doi:10.1038/nbt.4096 (2018).
- 29 Kang, H. M. *et al.* Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. *Nat Biotechnol* **36**, 89-94, doi:10.1038/nbt.4042 (2018).
- 30 Polański, K. *et al.* BBKNN: fast batch alignment of single cell transcriptomes. *Bioinformatics* **36**, 964-965 (2020).
- 31 Hie, B., Bryson, B. & Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nature biotechnology* **37**, 685-691 (2019).
- 32 Lotfollahi, M. *et al.* Mapping single-cell data to reference atlases by transfer learning. *Nat Biotechnol*, doi:10.1038/s41587-021-01001-7 (2021).
- 33 Wang, D. *et al.* iMAP: integration of multiple single-cell datasets by adversarial paired transfer networks. *Genome Biol* **22**, 63, doi:10.1186/s13059-021-02280-8 (2021).
- 34 Li, X. *et al.* Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nat Commun* **11**, 2338, doi:10.1038/s41467-020-15851-3 (2020).
- 35 Buttner, M., Miao, Z., Wolf, F. A., Teichmann, S. A. & Theis, F. J. A test metric for assessing single-cell RNA-seq batch correction. *Nat Methods* **16**, 43-49, doi:10.1038/s41592-018-0254-1 (2019).
- 36 Klimchenko, O. *et al.* A common bipotent progenitor generates the erythroid and megakaryocyte lineages in embryonic stem cell-derived primitive hematopoiesis. *Blood, The Journal of the American Society of Hematology* **114**, 1506-1517 (2009).

- 37 Trinchieri, G. Biology of natural killer cells. *Advances in immunology* **47**, 187-376 (1989).
- 38 LeBien, T. W. & Tedder, T. F. B lymphocytes: how they develop and function. *Blood, The Journal of the American Society of Hematology* **112**, 1570-1580 (2008).
- 39 Hoadley, K. A. *et al.* Multiplatform analysis of 12 cancer types reveals molecular classification within and across tissues of origin. *Cell* **158**, 929-944 (2014).
- 40 Vaswani, A. *et al.* in *Advances in neural information processing systems*. 5998-6008.
- 41 Cheng, J., Dong, L. & Lapata, M. Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733* (2016).
- 42 Traag, V. A., Waltman, L. & van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Sci Rep* **9**, 5233, doi:10.1038/s41598-019-41695-z (2019).
- 43 Becht, E. *et al.* Dimensionality reduction for visualizing single-cell data using UMAP. *Nature biotechnology* **37**, 38-44 (2019).
- 44 Ghader, H. & Monz, C. What does attention in neural machine translation pay attention to? *arXiv preprint arXiv:1710.03348* (2017).
- 45 Michel, P., Levy, O. & Neubig, G. Are sixteen heads really better than one? *arXiv preprint arXiv:1905.10650* (2019).
- 46 Vig, J. *et al.* Bertology meets biology: Interpreting attention in protein language models. *arXiv preprint arXiv:2006.15222* (2020).
- 47 Mariathasan, S. *et al.* TGFbeta attenuates tumour response to PD-L1 blockade by contributing to exclusion of T cells. *Nature* **554**, 544-548, doi:10.1038/nature25501 (2018).
- 48 Zhang, X. *et al.* CellMarker: a manually curated resource of cell markers in human and mouse. *Nucleic Acids Res* **47**, D721-D728, doi:10.1093/nar/gky900 (2019).
- 49 Lawson, K. A. *et al.* Functional genomic landscape of cancer-intrinsic evasion of killing

- by T cells. *Nature* **586**, 120-126, doi:10.1038/s41586-020-2746-2 (2020).
- 50 Schiebinger, G. *et al.* Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell* **176**, 928-943. e922 (2019).
- 51 Reichardt, J. & Bornholdt, S. Statistical mechanics of community detection. *Physical review E* **74**, 016110 (2006).
- 52 Malkov, Y. A. & Yashunin, D. A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence* **42**, 824-836 (2018).
- 53 Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118-127 (2007).

Methods

Input preprocessing

The input sequence list of top-expressing genes was obtained via descending sorting. The input to *tGPT* was formulated as [$\langle s \rangle$, G_1 , G_2 , G_3 , ..., $\langle e \rangle$], where G_1 , G_2 and G_3 are gene symbols and $\langle s \rangle$ and $\langle e \rangle$ are two special tokens respectively added to the start and end of the input sequence. The input sequence is padded with special token $\langle pad \rangle$ if its length is less than a predefined value. The input sequence list is truncated if its length exceeds the predefined value. We evaluated a length of 64 and 128 in this study. The dictionary used by *tGPT* consists of 20706 protein-coding genes.

The architecture of *tGPT*

Embedding layer transforms the input gene symbols into a real-value matrix that carries the information on gene token embedding and position encoding. The gene token embedding was

obtained via an embedding layer (parameterized as W_e) that maps the indices of input genes obtained from the gene symbol dictionary to real-value space. The position encoding (parameterized as W_p) carries information on the sorted gene rankings. For an input sequence $U = \{G_{-k}, \dots, G_{-1}\}$, where k is the width of context window, the embedding layer injects position encoding onto gene token embedding as:

$$h_0 = UW_e + W_p$$

Transformer decoder blocks applies multi-headed masked self-attention over the input embeddings followed by position-wise feed-forward layers, then through a *softmax* layer. *tGPT* use a multi-layer of *transformer decoder*¹⁷.

$$h_l = \psi(h_{l-1}) \forall l \in [1, n]$$

$$P(U) = \phi(h_n W_t^T)$$

where ψ is the transformer decoder block and ϕ is the softmax layer, and W_t is the embedding matrix of the l^{th} decoder block.

Masked Self-Attention is a variant self-attention mechanism⁴⁰. Each attention head adopts the scale dot-product attention to map a query and a set of key-value pairs to an output. The input consists of query and key of dimension d_k , and value of dimensions d_v . Self-attention is calculated as dot products of the query (Q_i) with key (K_i) divided each by $\sqrt{d_k}$ and multiply with value (V_i) after *softmax* transformed⁴¹:

$$SelfAttn_i(Q_i, K_i, V_i) = softmax(\frac{Q_i K_i^T}{\sqrt{d_k}}) V_i$$

Masked self-attention is implemented with the aid of attention mask. It basically always scores the future tokens as 0 so *tGPT* cannot pick from future. The multi-head self-attention is

formulated as:

$$MultiHead(Q, K, V) = Concat(SelfAttn_1, \dots, SelfAttn_h)W^O$$

where $W^O \in \mathbb{R}^{hd_v \times d_{model}}$ denotes the learned output projection matrix.

Position-wise Feed Forward neural network is a layer with fully-connected feed-forward layer. This layer consists of two linear transformations with a *ReLU* activation function in between:

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

where W_1 and W_2 are weight matrices and b_1 and b_2 are the bias.

Training scheme

tGPT was pretrained with a batch-size of 64 for 100 epochs. We used Adam with $\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay of 0.01 and a learning rate of 0.003. The learning rate is warmed up for four epochs, and then decays to 0 following a cosine schedule¹². *tGPT* was trained with *PyTorch* (version 1.7.1) and *transformers* (version 4.10.0) on NVIDIA DGX A100 with 8 GPUs each with 40 Gb memory.

Clustering on feature representation from *tGPT*

We respectively extracted the feature representations from the embedding layer and 8 different transformer layers. The extracted features were used to construct K-Nearest Neighbors (KNN) graphs for subsequent community detection by Leiden algorithm⁴² implemented in *Scanpy* (version 1.8.1). We performed grid-search to identify optimal values of two parameters $n_neighbors$ and $resolution$ that are the most relevant for clustering. The value of $n_neighbors$ examined was ranged from 5 to 100 with step of 5. The value of $resolution$ examined was ranged from 0.1 to 2 with step of 0.2. The uniform manifold approximation and projection

(UMAP) visualization⁴³ is used.

Features derived from self-attention

Entropy of the self-attention matrices for a given input sequence is calculated as⁴⁴:

$$Entropy_{\alpha}(x_i) = -\sum_{j=1}^i \alpha_{i,j}(x) \log(\alpha_{i,j}(x)),$$

where α is the self-attention matrix and $\alpha_{i,j}$ is the attention weight between the i^{th} and j^{th} tokens. We averaged the entropy of all cells in a cluster to derive a cluster-level entropy.

Head importance score⁴⁵ is defined as the influence of input on head output. It is calculated via gradient backpropagation, formulated as:

$$I_h = \mathbb{E}_{x \sim X} \left| SelfAttn_h(x)^T \frac{\partial \mathcal{L}(x)}{\partial SelfAttn_h(x)} \right|$$

where x is the input sequence and $\mathcal{L}(x)$ is the corresponding loss given the input. I_h is high score while $SelfAttn_h(x)$ is liable to have a large effect on the model.

Token attribution score is defined as the norm of the learned token features (x_i) extracted from *tGPT*, which is defined as:

$$Attribution = \sqrt[2]{x_1^2 + x_2^2 + \dots + x_N^2}$$

Attention analysis in relation to signaling

We define an attention-based pathway signaling score in a similar way as⁴⁶:

$$p_{\alpha}(f) = \sum_{x \in X} \sum_{i=1}^x \sum_{j=1}^x f(i,j) \alpha_{i,j}(x) \bigg/ \sum_{x \in X} \sum_{i=1}^x \sum_{j=1}^x \alpha_{i,j}(x)$$

where $\alpha_{i,j}$ is the attention weight between the i^{th} and j^{th} gene. For a given gene signature, we

set $f(i, j) = 1$ if the i^{th} gene or j^{th} gene occurs in that gene set.

Dataset collection

We collected the transcriptomes of 22.3 million single-cells (**Supplementary Table 1**), 9318 bulk tissue transcriptomes of TCGA cohort from the supplemental data of pan-cancer immune landscape study²², 11,688 bulk tissue transcriptomes from *GTEX* database²¹ and 298 bulk tissue transcriptomes from the clinical trial study on immunotherapy for urothelial carcinoma⁴⁷. We discarded mitochondrial genes, ribosomal genes and non-protein coding genes for the single-cell data. Four single-cell and two bulk tissue sequencing datasets are used in downstream evaluation of *tGPT*:

Human Cell Atlas Census of Immune Cells (HCA) - Bone marrow cells ($n = 282,588$) from 64 healthy donors in *Human Cell Atlas (HCA)* project. The data are subjected to 10x sequencing protocol¹⁸ and contained 18 cell types such as hematopoietic stem cells (HSCs), mesenchymal stem cells (MSCs), erythrocytes, megakaryocytes and different kinds of immune cells.

Human cell Landscape (HCL) - *HCL* dataset includes 586,135 human cells obtained from a Chinese Han population¹⁹, the dataset encompasses samples of fetal and adult tissue and covered 60 human tissue types, and are subjected to Microwell-seq protocol.

Tabula Mursi - The *Tabula Muris* dataset ($n = 54,865$) is consisted of single-cells sorted by FACS from Mouse Cell Atlas⁴ across 20 different organs subjected to 10x and Smart-seq2 sequencing protocols.

The Cancer Genome Atlas (TCGA) - The *TCGA* dataset is consisted of 9,318 bulk samples with primary cancer and matched normal samples spanning 33 cancer types.

Genotype-Tissue Expression Project (GTEx) - The *GTEx* dataset includes 11,688 bulk samples across 30 organs obtained from healthy donors.

Known marker genes of different cell types are curated from CellMarker database⁴⁸: plasma

cell (*MZB1*), DCs and monocytes (*CST3*, *FCER1A*, *IRF7*, *CD14* and *FCGR3A*), megakaryocyte (*PPBP* and *PF4*), B cell (*CD79A*, *CD79B* and *MS4A1*) NK and cytotoxic T cell (*NKG7*, *FGFBP2*, *GNLY*, *GZMA*, *GZMB* and *PRF1*).

Cell state signatures are curated from CellMarker database⁴⁸, including progenitor signaling (*STMN1*, *TUBA1B* and *HIST1H4C*), naïve signaling (*CCR7*, *LEF1* and *SELL*) and cytotoxic signaling (*GZMA*, *CD8A*, *CD8B*, *GZMB*, *PRF1*, *IL2*, *GNLY*, *GAMK*, *IFNG* and *NKG7*).

T cell infiltration signature is obtained from CellMarker database⁴⁸; it consists of *CD3D*, *CD3E* and *CD8A*.

Tumor evasion signature is curated from the Figure 1 of a previous study⁴⁹.

Diffusion pseudo-time maps construction

We constructed the diffusion pseudo-time maps using package *Pegasus*²⁴ (v1.4.3), and the cell trajectory was visualized with force-directed layout embedding (FLE) algorithm⁵⁰. We set δ and $n\delta$ to its default values: $\delta = 2.0$ and $n\delta = 5,000$.

Firstly, we used the features obtained from last transformer decoder blocks to construct affinity matrix of cells $W_{n \times n}$, and the top- k nearest neighbor cells were find by community detection algorithm⁵¹ and the HNSW algorithm⁵², and the formula of affinity matrix is define as:

$$K(x, y) = \left(\frac{2\sigma_x \sigma_y}{\sigma_x^2 + \sigma_y^2} \right)^{\frac{1}{2}} \exp\left(-\frac{\|x - y\|^2}{\sigma_x^2 + \sigma_y^2}\right) \quad (1)$$

$$k'(x, y) = \frac{K(x, y)}{q(x)q(y)} \quad (2)$$

$$W(x, y) = \begin{cases} k'(x, y), & y \in n(x) / x \in n(y) \\ 0, & otherwise \end{cases} \quad (3)$$

The formula (1) represented the distance between cell- x and cell- y , σ_x is the x 's local kernel

width, x and y are features of last transformer decoder block for cell- x and cell- y . The affinity matrix W was calculated as the density-normalized kernel according to formula (3).

We then calculated the Markov chain transition matrix P and the symmetric transition matrix Q as the formula:

$$D = \text{diag}(\sum_y W(x, y))$$

$$P = D^{-1}W, Q = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$$

The symmetrical matrix Q can be decomposed as UAU^T . Let $\Psi = D^{-\frac{1}{2}}U$. A family with parameter timescale of t for approximated diffusion maps $\{\Psi_t\}_{t \in \bullet \cup \{\infty\}}$ is defined as:

$$\Psi_t(x_i) = \begin{pmatrix} \lambda_1^t \Psi_1(i) \\ \lambda_2^t \Psi_2(i) \\ \vdots \\ \lambda_{n-1}^t \Psi_{n-1}(i) \end{pmatrix}$$

$$\Psi_t'(x_i) = \sum_{t'=1}^t \Psi_{t'}(x_i) = \begin{pmatrix} \lambda_1 \frac{1-\lambda_1^t}{1-\lambda_1} \Psi_1(i) \\ \lambda_2 \frac{1-\lambda_2^t}{1-\lambda_2} \Psi_2(i) \\ \vdots \\ \lambda_{n-1} \frac{1-\lambda_{n-1}^t}{1-\lambda_{n-1}} \Psi_{n-1}(i) \end{pmatrix}$$

Benchmark methods

We also performed single-cell analysis using *Scanpy* (version 1.6.0), *Pegasus* (version 1.4.3) and *scVI* (version 0.13.0). Batch-correction was performed with *MNN* (version 1.8.0)²⁶, *Combat* (version 1.8.0)⁵³, *Harmony* (version 0.1.6)²⁷, *Seurat* (version 3.1.5)^{28,29}, *Pegasus* (version 1.4.3)²⁴, *Scanorama* (version 1.7.1)³¹, *DESC* (version 2.1.1)³⁴, *iMAP* (version 1.0.0)³³,

scVI (version 0.13.0)⁸, *scArches* (version 1.7.0)³², *BBKNN* (version 1.7.1)³⁰.

Scanpy is a comprehensive toolkit for analyzing single-cell transcriptome. We first filtered out cells with the number of expressing genes < 200 or *mitochondrial counts* > 30%. We used the function *scanpy.pp.highly_variable_genes* to selected highly variable genes by setting *max_mean* to 3 and *min_mean* to 0.0125, which are the default values. We then applied clustering pipeline and grid-search to perform single-cell clustering on KNN graph. The UMAP is used for visualizing clustering result.

scVI is a deep generative model for mining the single-cell omics data. We filtered out cells with the number of expressing genes < 200 or *mitochondrial counts* > 30%, and selected HVGs with *scanpy.pp.highly_variable_genes* by setting *max_mean* to 3 and *min_mean* to 0.0125. We used the default parameter of *scVI* to extract the 10 latent features. These latent features were used to construct KNN graphs for community detection by Leiden algorithm⁴².

Pegasus is complete single-cell analysis pipeline that is efficient on large datasets. We used the recommended parameters: *min_genes* of 500, *max_genes* of 6000, and *percent_mito* of 10. We identified the robust genes with the default *percent_cells* of 0.05. Single-cell clustering was performed on KNN graph followed by Leiden algorithm⁴² for community detection.

Clustering and batch-effect metrics

We used Adjusted Rand Index (*ARI*), Normalized Mutual information (*NMI*) and Fowlkes-Mallows Index (*FMI*) to measure clustering performance. We used the *kBET* acceptance rate³⁵ as a measurement of batch-effect. The clustering metrics of *ARI*, *NMI* and *FMI* were calculated with sklearn (version 0.21.2). *kBET* acceptance rate is computed with *Pegasus* (version 1.4.3).

ARI is calculated based on the contingency table summarizing the truth labels and clustering, and the rows and columns represent truth and clustering labels in the contingency table, respectively. The formula is as follows:

$$ARI = \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{a_j}{2} \right] / \binom{n}{2}}{\frac{1}{2} \left[\sum_i \binom{a_i}{2} + \sum_j \binom{a_j}{2} \right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{a_j}{2} \right] / \binom{n}{2}}$$

where n_{ij} denoted the numbers of cell in common between clustering labels and truth labels, a_i the sum of i^{th} row and a_j the sum of j^{th} column of the contingency table.

NMI is also used to measure the similarity between the clustering labels and actual labels. We assumed that the clustering labels and actual labels of N cells are U and V , and the entropy of U and V is as the following formula:

$$H(U) = - \sum_{i=1}^{|U|} P(i) \log(P(i))$$

$$H(V) = - \sum_{j=1}^{|V|} P'(j) \log(P'(j))$$

where $p(i) = |U_i| / N$ is the probability that a cell picked at random from U falls into U_i , $p'(j) = |V_j| / N$ is the probability that a cell picked at random from V falls into V_j . We then calculated the mutual information (MI) between U and V , and normalized the mutual information:

$$MI(U, V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} P(i, j) \log\left(\frac{P(i, j)}{P(i)P'(j)}\right)$$

$$NMI(U, V) = \frac{MI(U, V)}{\sqrt{H(U)H(V)}}$$

where $p(i, j) = |U_i \cap V_j| / N$ is the probability that a cell picked at random falls into classes U_i and V_j .

Fowlkes-Mallows Index (FMI) is used to measure the consistency between clustering results and real category, and the range of index is from 0 to 1. The *FMI* metric is denfined as the geometric mean between of the precision and recall:

$$FMI = \frac{TP}{\sqrt{(TP + FP)(TP + FN)}}$$

where *TP* is true positive, *FP* false positive, *FN* false negative.

kBET acceptance rate is a measurement of batch effect. We assumed that the dataset of cells with batches of m , and there are n_j cells in batch j . The batch mixing frequency denotes as:

$$f = (f_1, \dots, f_m)$$

where $f_j = \frac{n_j}{N}$. Then, we calculated the number of neighbors of cell- i belonging to batch j is

n_{ji}^k . Its χ^2 test statistic and p -value with degrees of $(m-1)$ are defined as follows:

$$k_i^k = \sum_{j=1}^m \frac{(n_{ji}^k - f_j \cdot k)^2}{f_j \cdot k}$$

$$p_i^k = 1 - F_{m-1}(k_i^k)$$

where $F_{m-1}(x)$ represents the cumulated density function. The *kBET* acceptance rate is defined as the percentage of cells that accept the null hypothesis at significance level α as follows:

$$kBET-rate = \frac{\sum_{i=1}^N I(p_i^k \geq \alpha)}{N} \times 100\%,$$

$I(x)$ is the indicator function where $I(x) = 1$ if $x > 0$ otherwise $I(x) = 0$. We used *Pegasus* (v1.4.3) to calculate the *kBET* acceptance rate by setting K and α to 5 and 0.01, respectively.

Figure legends

Figure 1. A flowchart illustrating the framework of tGPT and its downstream applications.

It consists of three components: development of tGPT, applications of tGPT for single-cell and bulk tissue transcriptomes.

Figure 3. Distinct features of different cell types from the HCA dataset learned by tGPT.

(A) Heatmap representation of attention head entropy for different cell types. (B) Heatmap representation of attention head importance for different cell types. (C) Dot plot illustrating the attribution scores for cell type specific genes. (D-G) Scatter plots illustrating the distribution

692

pro-B cells to plasma cells. **(E-G)** Cell state signatures for progenitor signaling, naïve signaling and cytotoxic signaling. **(H-K)** The diffusion map of *HCL* dataset and its main branches.

Figure 5. The association of features learned by *tGPT* versus genomic alteration events and clinical phenotype. (A) Heatmap representation of attention head importance score across different cancer types on the TCGA dataset. Association of attention head entropy versus tumor

mutation burden **(B)**, *TP53* mutation **(C)**, homologous recombination deficiency **(D)**, genome doubling **(E)** and overall survival **(F)** on the TCGA cohort. **(G and H)** Heatmap representation of attention head importance and entropy on the urothelial carcinoma stratified by RECIST response. CR, complete response; PR, partial response; SD, stable disease; PD, progress disease. **(I)** The varying entropy patterns from SD to PR to CR with PD as baseline. **(J)** Exemplified violin plots depicting attention head entropy in SD/PD versus CR/PR. **(K)** The varying of patterns of tumor evasion signature from SD to PR to CR with PD as baseline. **(L)** The varying of patterns of T cell infiltration signature from SD to PR to CR with PD as baseline. **(M)** Association between attention head entropy and overall survival on the urothelial carcinoma dataset. **(N)** Exemplified survival curves stratified by attention head entropy.

Supplementary Figure 1. Radar charts illustrating the clustering performance achieved by feature representations extracted from different layers of *tGPT* for the top 62 (A) and 126 (B) expressing genes on *HCA*, *HCL*, *Tabula Muris*, *Macaque Retina*, *GTEX*, and *TCGA* datasets.

Supplementary Figure 2. The clustering performance with grid search for resolution and number of neighbors for the top 62 (A) and 126 (B) expressing genes among feature representations extracted from different layers on the *HCA* dataset. Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 3. The clustering performance with grid search for resolution and number of neighbors for the top 62 (A) and 126 (B) expressing genes among feature representations extracted from different layers on the *HCL* dataset. Contour maps depict

different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 4. The clustering performance with grid search for resolution and number of neighbors for the top 62 (A) and 126 (B) expressing genes among feature representations extracted from different layers on the *Tabula Muris* dataset. Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 5. The clustering performance with grid search for resolution and number of neighbors for the top 62 (A) and 126 (B) expressing genes among feature representations extracted from different layers on the *Macaque Retina* dataset. Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 6. The clustering performance with grid search for resolution and number of neighbors for the top 62 (A) and 126 (B) expressing genes among feature representations extracted from different layers on the *GTEX* dataset. Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 7. The clustering performance with grid search for resolution and number of neighbors for the top 62 (A) and 126 (B) expressing genes among feature representations extracted from different layers on the *TCGA* dataset. Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution*

and *N-neighbors*.

Supplementary Figure 8. UMAP visualization on different datasets obtained from different methods.

Supplementary Figure 9. The full annotation of UMAP visualization of different methods on the *HCL* dataset. The *NMI* metric and annotation of cells are shown.

Supplementary Figure 10. The full annotation of UMAP visualization of different methods on the *Tabula Muris* dataset. The *NMI* metric and annotation of cells are shown.

Supplementary Figure 11. Radar charts illustrating the best clustering metrics for different methods across different datasets obtained from grid search. *ARI*, Adjusted Rand Index; *NMI*, Normalized Mutual information; *FMI*, Fowlkes-Mallows Index.

Supplementary Figure 12. The clustering performance with grid search for resolution and number of neighbors for *Scanpy* (A), *Pegasus* (B), and *scVI* (C) on the *HCA*, *HCL*, *Tabula Muris* and *Macaque Retina* dataset. Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 13. The UMAP visualization plots of different batch-correction methods on the *HCA* dataset (A to K) and *kBET* acceptance rate (L).

Supplementary Figure 14. Radar charts illustrating the best clustering metrics for different batch-correction methods obtained from grid search on the *HCA* dataset. *ARI*,

Adjusted Rand Index; *NMI*, Normalized Mutual information; *FMI*, Fowlkes-Mallows Index.

Supplementary Figure 15. The clustering performance with grid search for resolution and number of neighbors for different batch-correction methods on the *HCA* dataset.

Contour maps depict different cluster metrics (i.e. *NMI*, *ARI* and *FMI*) with respect to different values of *Resolution* and *N-neighbors*.

Supplementary Figure 16. The survival curves stratified by attention head entropy across multiple cancer types on TCGA dataset. ACC, Adrenocortical carcinoma; CHOL, Cholangiocarcinoma; KIRC, Kidney renal clear cell carcinoma; READ, Rectum adenocarcinoma; KIRP, Kidney renal papillary cell carcinoma; LGG, Brain Lower Grade Glioma; MESO, Mesothelioma.

Supplementary Table 1. Source of data list.