Stanford University

清華大學 Tsinghua University

Single cell multi-modal integrative analysis with autoencoder

Team: Amateur

Qiao Liu, Wanwen Zeng — Stanford University

Chencheng Xu — Tsinghua University

What we did in the NeurIPS single cell competition

- We designed two autoencoder models to participate in Joint embedding and Modality prediction tasks

- Results
  - Rank 1st in Joint embedding track of both Multiome and CITE-seq (with pretrain)
    - Mean metrics: 0.8039 for CITE-seq, 0.8424 for multiome
  - Rank 2rd in ATAC2GEX subtask in Modality prediction
    - RMSE: 0.2266

Dataset for Joint Embedding task

- Multiome
  - Phase 1: **22463** cells
    - s1d1: 5616, s1d2: 6069, s2d1: 3811, s2d4: 5456, s3d6: 1511
  - Phase 2 training: **42492** cells (same as Phase1-v2)
    - s1d1: 5616, s1d2: 6069, s1d3: 3875, s2d1: 3811, s2d4: 5456, s2d5: 4395, s3d10: 3909, s3d3: 1496, s3d6: 1771, s3d7: 6094
- CITE-seq
  - Phase1: **43890** cells
    - s1d1: 4721, s1d2: 4451, s2d1: 9353, s2d4: 5026, s3d6: 9977, s3d7: 10362
  - Phase2 training: **66175** cells (same as Phase1-v2)
    - s1d1: 4721, s1d2: 4464, s1d3: 5484, s2d1: 9353, s2d4: 5026, s2d5: 8206, s3d1: 8582, s3d6: 9977, s3d7: 10362

Single cell Joint embedding with autoencoder

- Autoencoder with latent feature regularization

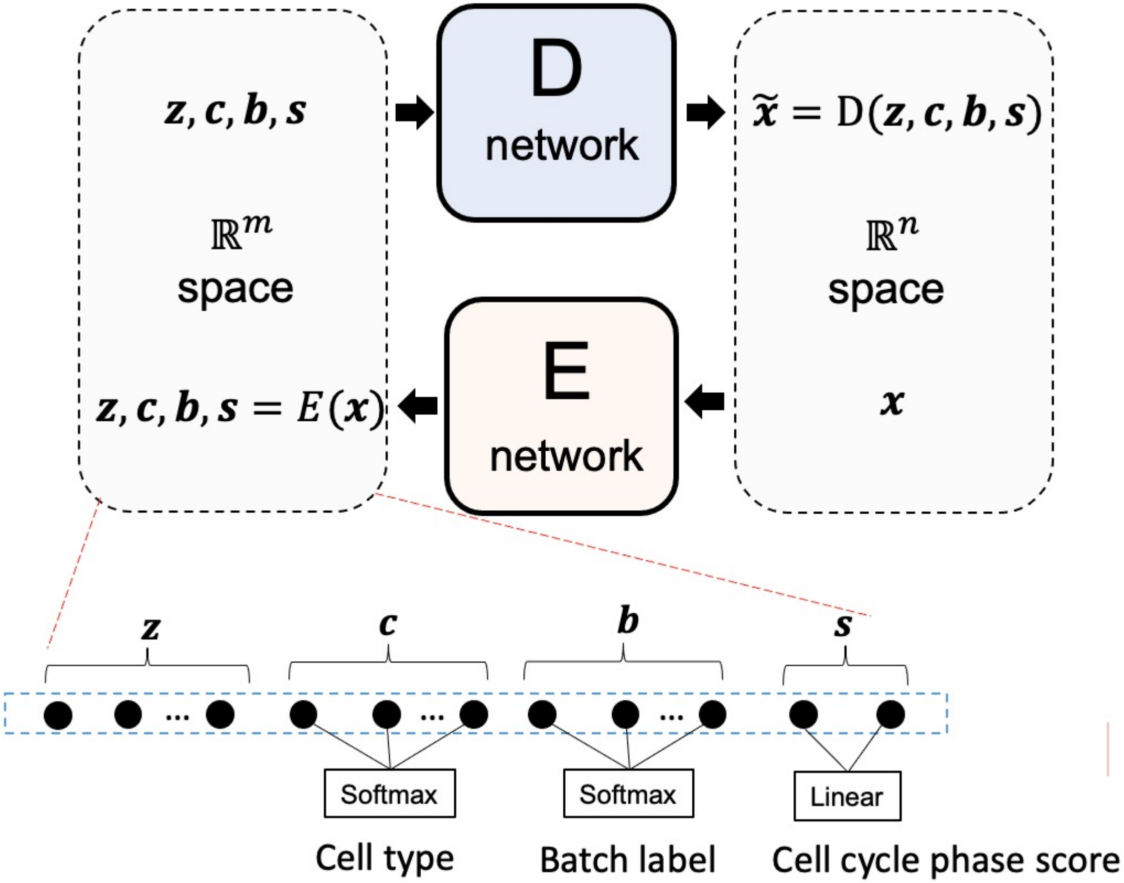
- First applied SVD to each modality (100 dim), then concatenate them and fed to an autoencoder model

- Autoencoder model aims at learn a low-dimensional representation in the latent space

- In the mean while, we desire that the latent feature could predict cell type, batch id and cell cycle phase score (S and G2M)
    - For batch label, we match the distribution with a Uniform distribution (for eliminating batch effect)

Model architecture

- Modified from scDEC model (*Nat Mach Intell 3, 536–544, 2021*) for single cell representation learning
- Compared to scDEC, we remove all discriminators, and add additional constrains in the latent space

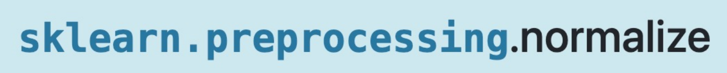


$\boldsymbol{z}, \boldsymbol{c}, \boldsymbol{b}, \boldsymbol{s}$
D
network
$\tilde{\boldsymbol{x}} = \mathrm{D}(\boldsymbol{z}, \boldsymbol{c}, \boldsymbol{b}, \boldsymbol{s})$
$\mathbb{R}^m$
space
$\mathbb{R}^n$
space
E
network
$\boldsymbol{z}, \boldsymbol{c}, \boldsymbol{b}, \boldsymbol{s} = E(\boldsymbol{x})$
$\boldsymbol{x}$
$\boldsymbol{z}$
$\boldsymbol{c}$
$\boldsymbol{b}$
$\boldsymbol{s}$
Softmax
Softmax
Linear
Cell type
Batch label
Cell cycle phase score

Data preprocessing

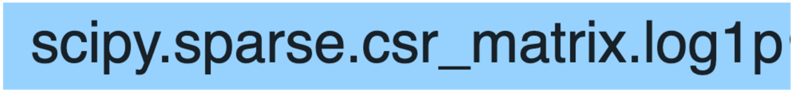
- We started with the raw reads count (.layers[“counts”])

- Step1: L1-normalize the across cells (normalize sequencing depth)

API: 

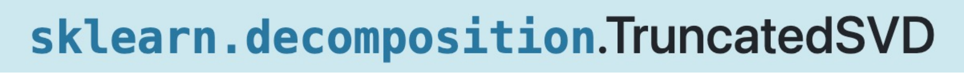
sklearn.preprocessing.normalize

- Step2: Scaling (10^4) and $\log_{10}(1+x)$ normalized

API: 

scipy.sparse.csr_matrix.log1p

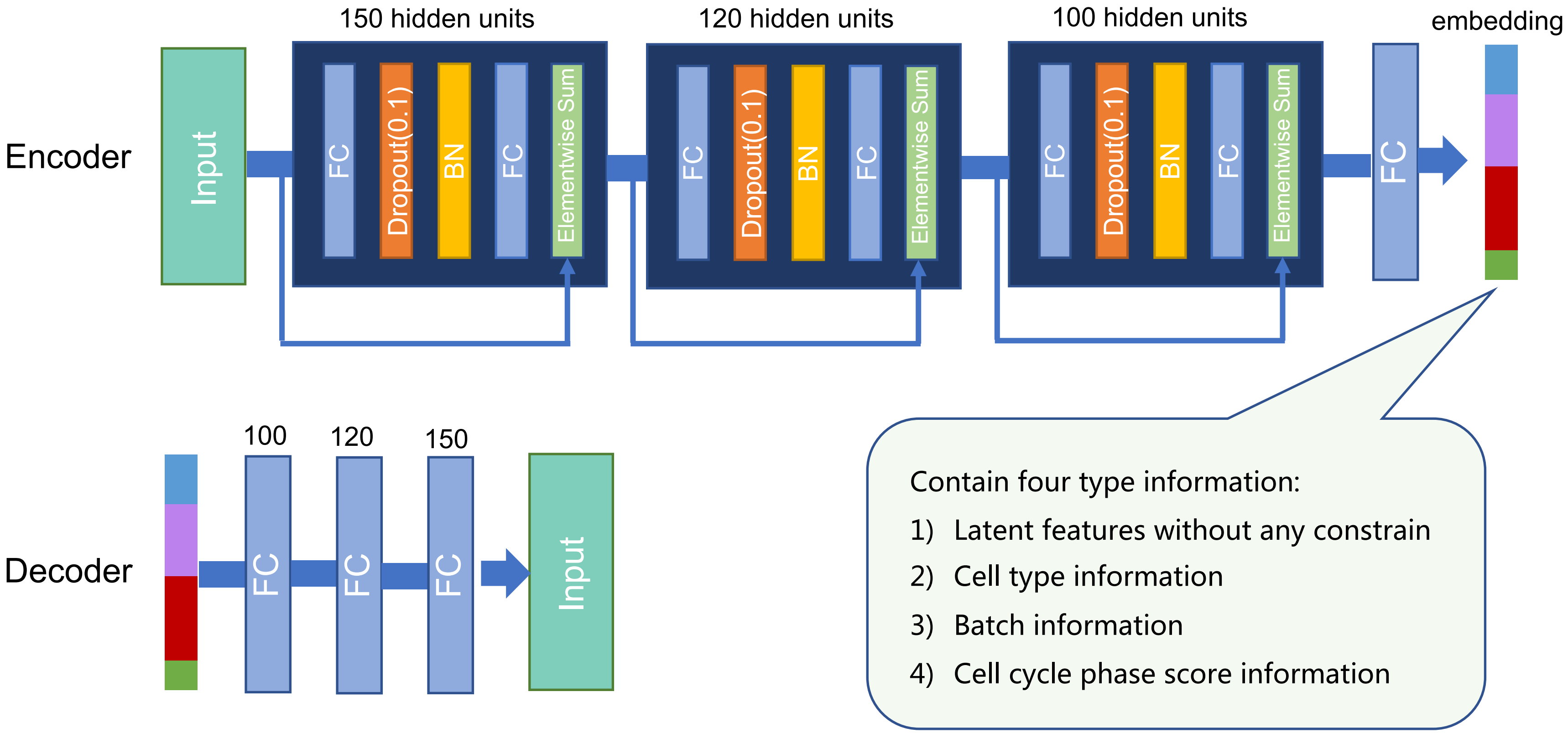
- Step3: SVD transformation, reduced to 100 dimension for each modality (except ADT)

API: 

sklearn.decomposition.TruncatedSVD

Hyperparameter setting

- For encoder, we use fully connected layers with residual connections, fully connected layers were used for decoder
- For Multiome, embedding dim is set to 33 (5+21+5+2)
- For CITE-seq, embedding dim is set to 58 (5+45+6+2)



Contain four type information:

1) Latent features without any constrain
2) Cell type information
3) Batch information
4) Cell cycle phase score information

Model pretrain

- Three predictors in the latent space for predicting cell type, batch id, and cell cycle phase score, respectively

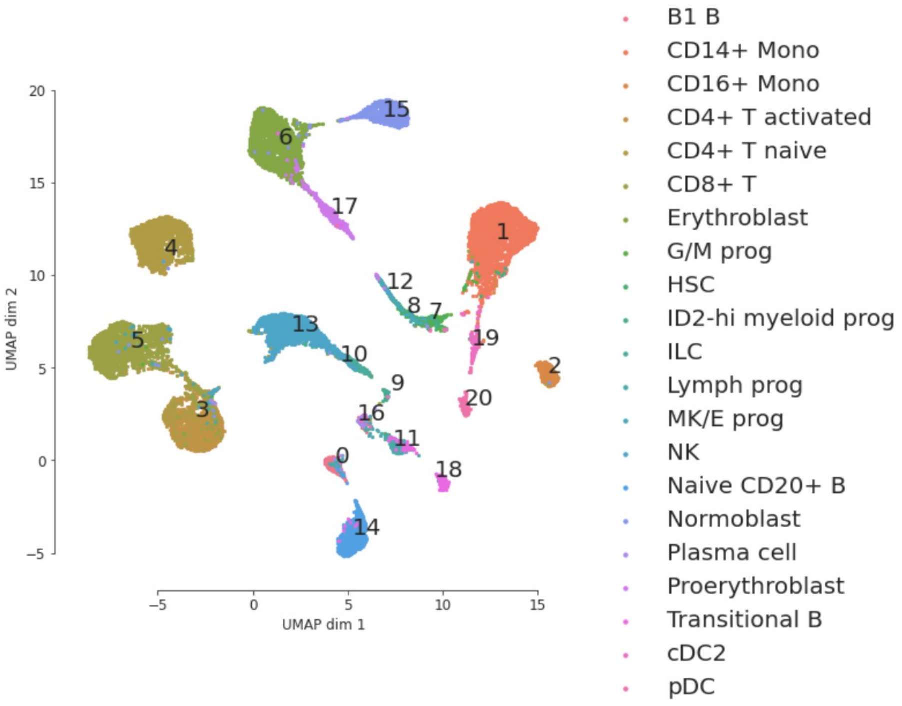
- For the exploration data, containing cell type, and batch information, the AE losses have four loss terms.
  - $loss_{rec}$: reconstruction loss
  - $loss_{ce\_c}$: cross entropy loss for cell type
  - $loss_{ce\_b}$: cross entropy loss for batch
  - $loss_{phase}$: MSE loss for cell cycle phase score

- To eliminate batch effect, we want the classifier to be as random as possible
  - Instead of using true batch label, we used a uniform distribution instead.

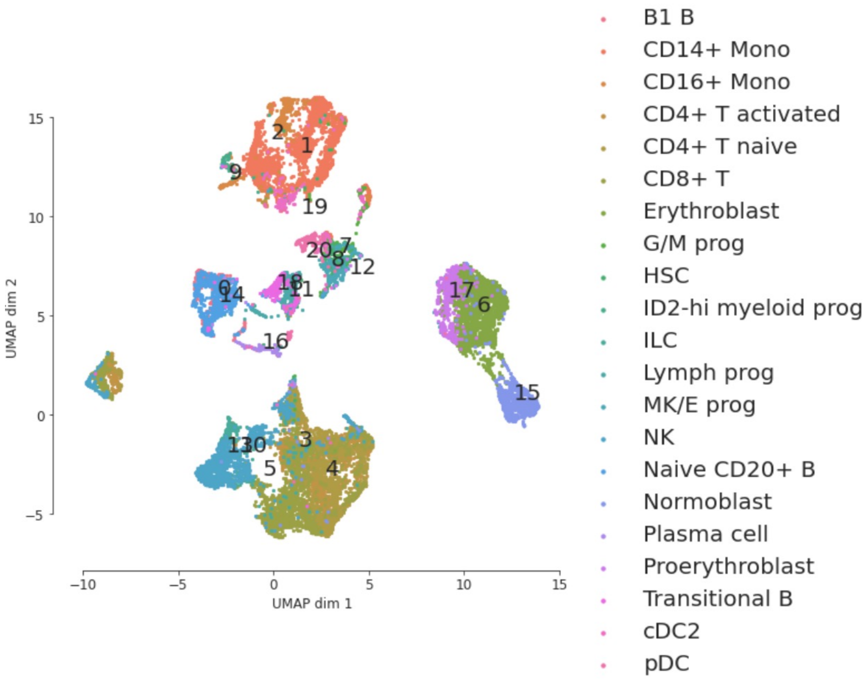
Total loss: $loss = 0.7\ loss_{rec} + 0.2\ loss_{ce\_c} + 0.05\ loss_{ce\_b} + 0.05\ loss_{phase}$

Pretrain visualization results

- Joint embedding for the JAE with Multiome exploration data
- Baseline method: Only SVD and concatenation



(our method)



(baseline method)

Online fine-tune

- Fine tune strategy:
  - Only fine tune with the AE reconstruction loss $loss_{rec}$

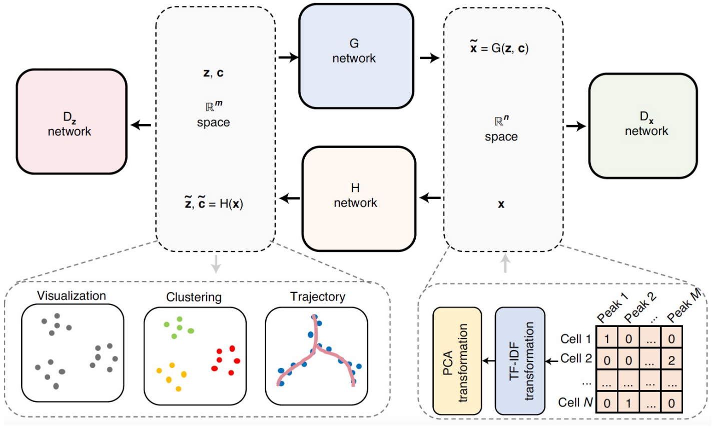
- For Multiome
  - Set a smaller learning rate (from 10^{-4} to $2*10^{-5}$)
  - We finetune online test data for 2 epochs

- For CITE-seq
  - We finetune online test data for only 1 epoch

Summary

- Pros
  - Easy and flexible to incorporate the annotation information (e.g., cell type label) to achieve a better embedding
- Cons
  - The dimension of latent feature directly relates to meta data (e.g., number of cell types)

- A more complicated version of JAE (with adversarial training) could be found in our recent work (scDEC, *Nat Mach Intell 3, 536–544, 2021*)



	Peak 1	Peak 2	...	Peak M
Cell 1	1	0	...	0
Cell 2	0	0	...	2
...	...	...	...	...
Cell N	0	1	...	0

Acknowledgement

Prof. Wing Wong

Assoc. Prof. Rui Jiang

Postdoc
Wanwen Zeng

PhD student
Chencheng Xu

- I would like to thank the organizers for their continuous support and quick response in discord, especially Robrecht Cannoodt

- Contact: liuqiao@Stanford.edu

- Project URL: https://github.com/kimmo1019/JAE