Single-cell RNA sequencing data meets Autoencoders

Dzvenymyra-Marta Yarish[1]

[1]Institute of Computer Science, University of Tartu, Tartu, Estonia.

Abstract

Recent advancements in single-cell sequencing technologies have enabled the generation of extensive datasets comprising millions of cells. Integrating such datasets is crucial for conducting meta-analyses to identify associations between cell states and patient conditions. However, the batch effect, arising from varied experimental protocols, poses a challenge to data integration. In this study, we addressed this challenge by applying the latest Autoencoder-based approach by De Donno *et al* [1] to integrate scRNA-seq data from three datasets of gene expression profiles from peripheral blood mononuclear cells.

1 Introduction

In recent years, the progress made in experimental and computational single-cell sequencing technologies has facilitated the creation of comprehensive datasets that include information from millions of cells. Such large-scale datasets give the opportunity to conduct so-called meta-analyses, which look for associations between cell states and different conditions such as diseases and lifestyle choices. Obtained data can in turn be used to understand which factors explain variation in gene expression at the cell level. The success of such studies depends on the quality of data integration between datasets, which has to correct for the batch effect between them. Those batch differences arise due to different experimental protocols and conditions, essentially adding noise to the original biological signal. Considerable efforts have been put to solving data integration problem for single-cell RNA sequencing (scRNA-seq) datasets. Various methods ranging from statistical to graph-based and deep learning models have been investigated in relation to this problem.

Autoencoder models are a class of artificial neural networks that have gained significant attention in recent years due to their ability to learn efficient representations of high-dimensional data. An autoencoder consists of an encoder network that compresses the input data into a lower-dimensional latent space, and a decoder network that reconstructs the original data from the latent representation. By training the model to minimize the reconstruction error, the autoencoder learns to capture the most salient features of the input data. Autoencoders have been successfully applied in various scientific domains, including genomics, image analysis, and natural language processing, offering valuable insights into the underlying structure and patterns of complex datasets.

The goal of this study is to apply the latest Autoencoder-based approach by De Donno *et al* [1] for scRNA-seq data integration to three datasets, containing gene expression data from peripheral blood mononuclear cells.

2 Methods

2.1 Autoencoder architecture

The model used in this work is a Conditional Variational Autoencoder (CVAE), which is widely utilized for data integration and perturbation modeling in single-cell genomics. It combines deep neural networks with latent variable modeling to learn a compressed representation of the input data and generate similar samples. The CVAE includes conditioning variables in both the encoder and decoder networks, enabling control over the generation process based on specific attributes or conditions. Furthemore, adding the condition to the input of CVAE can remove the unwanted source of variation from the data. During training, CVAEs optimize the reconstruction loss to capture essential features of the data and a regularization term, typically based on the Kullback-Leibler (KL) divergence, to structure the latent space.

When it comes to applications of CVAEs to RNA-seq data, it is a well-established practice to use a different reconstruction loss than mean-squared error between input and output. Namely, reconstruction error is defined as the likelihood of the distribution of the noise model instead of reconstructing the input data itself. Most common distribution is a negative binomial distribution with or without zero inflation [2].

De Donno *et al* proposed two novel additions to this standard model: *conditional embeddings* and *cell prototypes*. Conditional embeddings are learnable representations of conditions (e.g. samples or datasets.) These embeddings, of fixed dimensionality, are concatenated to the input data during training. Unlike one-hot-encoded vectors, these learnable conditional embeddings offer scalability in scenarios with a large number of conditions, as the dimensionality of the embeddings remains constant regardless of the number of conditions. Prototypes refer to representations of cell types created by averaging the gene expression of cells belonging to each specific cell type. The model learns to consolidate the latent representation of cells around their prototypes, leading to better preservation of biological signals. The distance of each cell to its closest prototype is used for cell-type classification, uncertainty estimation, and identification of unknown cell types. Moreover, prototypes enable the expansion of a reference atlas with novel cell types from labeled query data without the need for retraining the reference.

The architecture of the CVAE used in this report is as follows: both encoder and decoder are fully-connected neural networks with 3 hidden layers each of size 128; the conditional embedding dimension is 20, the latent space dimension is 30. Implementation can be found here: `https://github.com/theislab/scarches/` .

2.2 Data integration workflow

The steps for a data integration workflow can be summarized as follows:

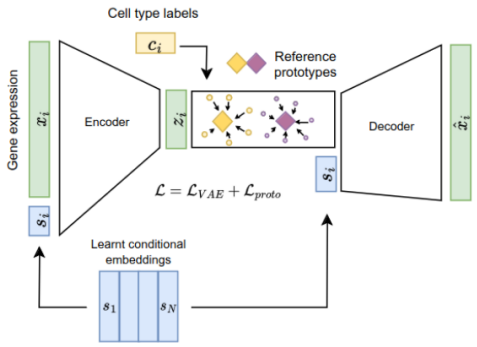
1. Reference building:

   Initialize N embeddings (N represents the number of studies in the reference) and optimize the reconstruction loss (L_{CVAE}) on the reference dataset. Then, initialize cell type prototypes and optimize the L_{CVAE} along with the prototype loss ($\eta L_{prototype}$) term (η is a hyperparameter). Store the learned prototypes with the model.

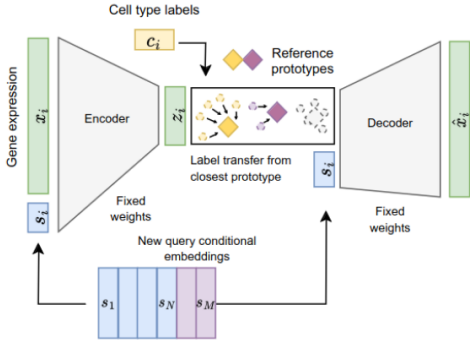
2. Query mapping:

   Freeze the weights of the encoder and decoder networks from the reference model. Initialize M additional learnable embeddings (M is the number of samples(batches) in the query datasets) and optimize the L_{CVAE} on the query data. Then, after unsupervised clustering of the latent representation of the query data, initialize unlabeled prototypes. Optimize the L_{CVAE} along with the $\eta L_{prototype}$. If all cells in the query are unlabeled, the learning objective is reduced to L_{CVAE} only.

Visually this process is illustrated in Figure 1.



(a) **Reference building**. The model is trained to reconstruct samples from reference datasets and create tight clusters for different cell types. It also learns conditional embeddings and constructs cell type prototypes.



(b) **Query mapping**. The model learns conditional embeddings for query datasets. Model weights are frozen, except for the embedding layers.

Figure 1: Data integration workflow, which consists of two stages. Image source: De Donno *et al.*

2.3 Datasets

In this study, we used three datasets of single cell scRNA-seq data from peripheral blood mononuclear cells - OneK1K [3], Perez [4] and Randolph [5]. Their statistics are summarized in Table 1. We applied the following data preprocessing to all three datasets: normalization of the count matrices and selection of 10,000 most highly variable genes. Those genes were identified in OneK1K data and then the same genes were selected from Perez and Randolph.

Table 1: Dataset statistics.

Dataset	Number of cells	Number of batches	Number of cell types	Condition
OneK1K [3]	1,267,768	75	29	-
Perez [4]	1,263,676	66	11	lupus
Randolph [5]	235,161	30	11	influenza infection

3 Results

3.1 Exploring OneK1K labels

We started our experiments with the largest OneK1K dataset. We divided it in half into reference and query sets. Then, at first step, we trained the whole network to reconstruct gene expression data from the reference set. At second step, we froze the weights of the CVAE and trained only the embedding layer responsible for encoding experimental batch information on the query set with the same reconstruction objective. To quantify the quality of learned representations, we explored the classification potential of the model. As was already described in Section 2, the prototypes for different cell types can be used to predict the cell type of a particular sample by finding the closest prototype in the latent space. We obtained almost identical classification accuracies for reference(0.63) and query sets (0.64), which means that the proposed setup with freezing of the AE weights is indeed working and can successfully integrate data from different batches. We investigated the predicted cell types in more details by building a confusion matrix (Figure 2). Information about which cell types are most often confused with each other can be used to refine cell types labels, merge types together or remove too general categories which is an important step in building large-scale consolidated cell atlases.

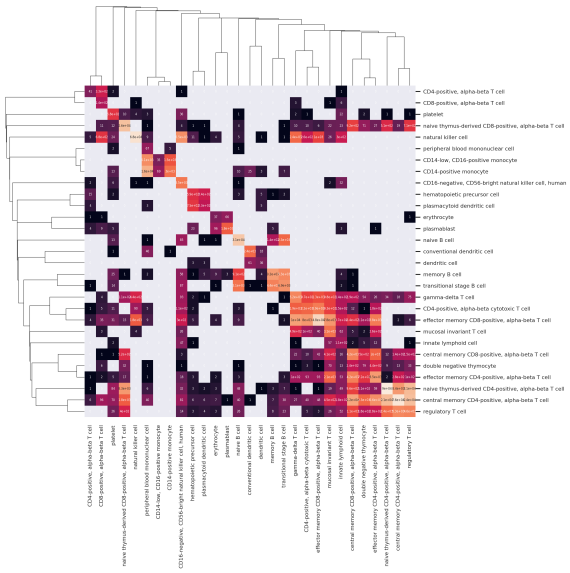
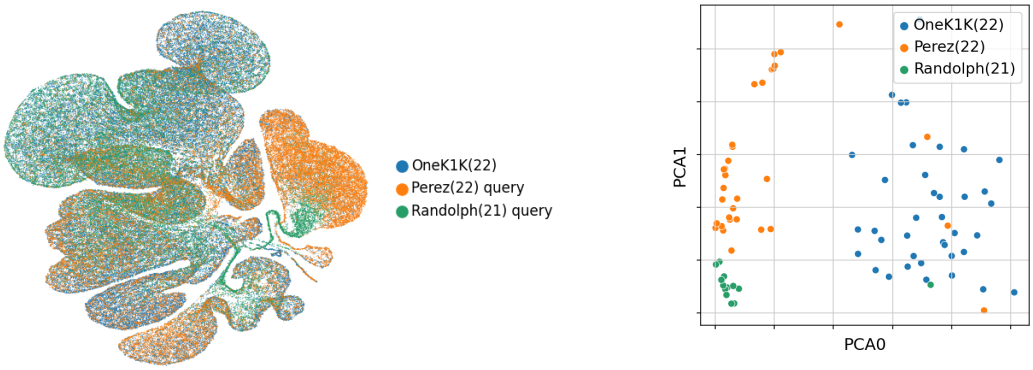


Figure 2: Confusion matrix of cell types classification results in OneK1K dataset.

3.2 Query mapping of Perez and Randolph

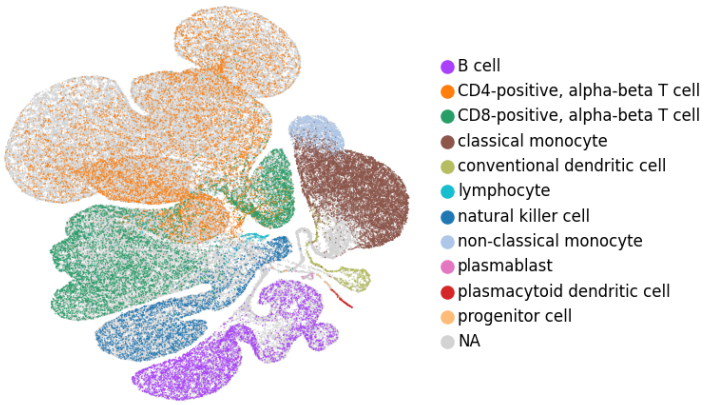
After building reference with OneK1K data, we mapped two other datasets, Perez and Randolp into the same latent space. Figure 3b shows the first two principal components of conditional embeddings for samples from three datasets. Those embeddings can later be studied on their own to better understand the relationship between technical and phenotypic factors. From Figure 3 we can conclude that the integration was successful and now meta-analysis can be conducted on three datasets.



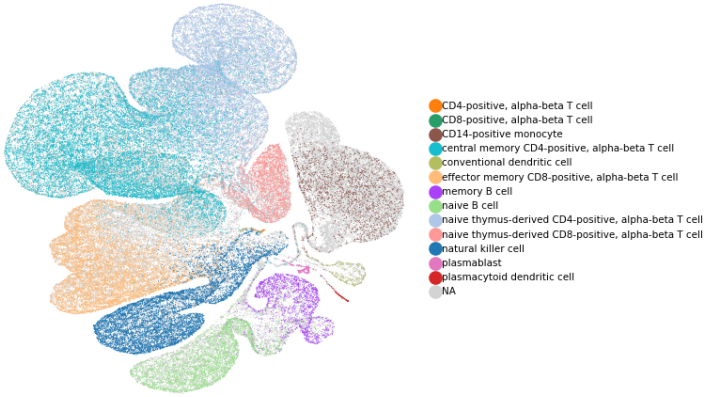
(a) (b)

Figure 3: Integration of three datasets.(a) U-map of latent cell representations, colored by dataset. (b) First two principal components of batch embeddings.

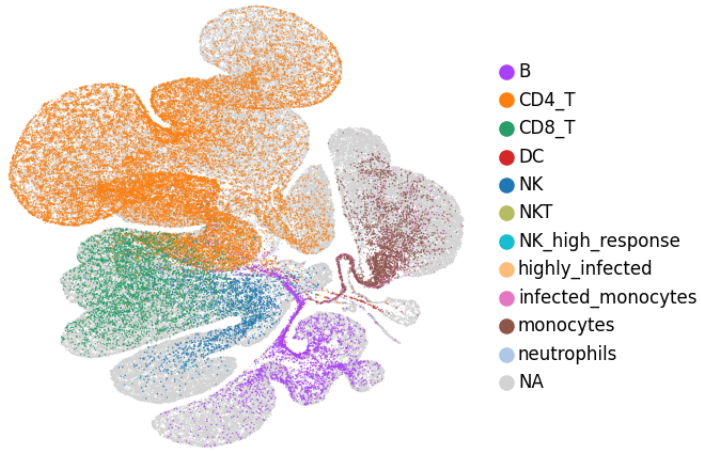
Figure 4 shows how the cells are clustered according to cell type labels from Perez (Figure 4a), OneK1K (Figure 4b) and Randolph (Figure 4c). We can observe that cell types are matched almost perfectly between Randolph and Perez, except for the fact that the small cluster of CD8-positive T cells from Perez is labeled as CD4-positive T cells according to Randolph. On the contrary, OneK1K offers more narrow categories of cell types, and this information can be used to label more precisely cells in Perez and Randolph datasets.



(a) Cell types in Perez(22) dataset. NA are cells from other two datasets.



(b) Cell types in OneK1K(22) dataset.

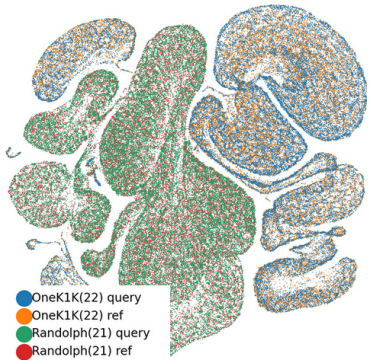


(c) Cell types in Randolph(21) dataset.

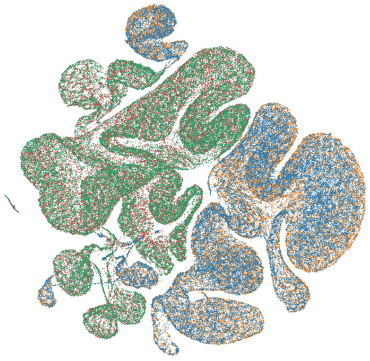
Figure 4: U-map of cell representations from 3 datasets, integrated into common latent space.

3.3 Using Randolph as a reference

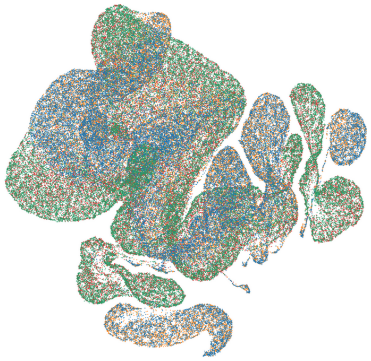
We also tested a hypothesis that adding Randolph dataset to the reference set will improve integration, as the Autoencoder will learn to create more unified representations for OneK1K and Randolph cells. As Randolph cell types don't map to OneK1K cell types in a straightforward way, we assessed three strategies for handling Randolp cell types during reference building: add them as is, ignore them and leave the cells unlabeled and unify some of the cell type labels between Randolph and OneK1K. As can be seen from Figure 5, first two strategies resulted in no integration at all, while the third one left some of isolated clusters of cells that belong exclusively to one dataset. Therefore, using one dataset only as reference and then adding new ones as query datasets seems to be the best strategy.



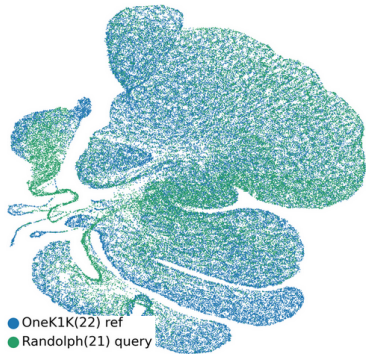
(a) Ignore Randolph cell types and add Randolph data to reference.



(b) Add all Randolph cell types.



(c) Manually merge some of the cell types between OneK1K and Randolph.



(d) Ignore Randolph cell types and don't add Randolph data to reference.

Figure 5: U-maps of cell representations from OneK1K and Randolph, using different cell types integration strategies.

4 Discussion

First of all, it is crucial to note the importance of data processing for such studies. Apart from the fact that the count data has to be normalized with one of the well-established methods, it is

critical to properly clean the dataset, removing entries which don't contain any meaningful RNA and expression data. Secondly, evaluating of the quality of data integration is a challenging task. Visual assessment of plots created using U-map or any other dimensionality reduction techniques can only provide a general idea of how good the integration is. However, if we want to compare several integration methods, visual evaluation cannot provide reliable results. A number of metrics was developed to test the integration quality, but they only work in conjunction with well-known benchmark datasets and aren't suitable for novel data, that hasn't been extensively analyzed before. In conclusion, we cannot right now say how effective was the integration of three datasets that we performed. It depends on the results of further analysis of integrated data and whether it will be able to detect any meaningful associations.

5 Conclusion

In conclusion, this study utilized the recent approach proposed by De Donno *et al* [1] in order to integrate scRNA-seq data from three datasets comprising gene expression profiles derived from peripheral blood mononuclear cells. The obtained latent cell representations can be used for further analysis.

References

[1] C. D. Donno *et al.*, "Population-level integration of single-cell datasets enables multi-scale analysis across samples," *bioRxiv*, 2022.

[2] G. Eraslan, L. M. Simon, M. Mircea, N. S. Mueller, and F. J. Theis, "Single-cell rna-seq denoising using a deep count autoencoder," *Nature Communications*, vol. 10, 2019.

[3] S. Yazar *et al.*, "Single-cell eqtl mapping identifies cell type–specific genetic control of autoimmune disease," *Science*, vol. 376, 2022.

[4] R. K. Perez *et al.*, "Single-cell rna-seq reveals cell type–specific molecular and genetic associations to lupus," *Science*, vol. 376, 2022.

[5] H. E. Randolph *et al.*, "Genetic ancestry effects on the response to viral infection are pervasive but cell type specific," *Science*, vol. 374, pp. 1127–1133, 2021.