slide 1:

Hi everyone, this is Songyang. It is my first time making research presentation, so I am a little nervous. If I do not make myself clear, please feel free to ask me questions. Today, my talk is about embedding co-assay data in the same space. The intuition for such embedding is that now we have both sc-RNA and sc-ATAC seq data for the same cells, we wonder if there are interesting relationship between peaks, genes and cells.

slide 2:

The data we analyzed on comes from Prof.Chen’s paper, adult and neonatal mouse brain cortex data. For the simplicity, I will just call them ad data and p0 data later in my talk.

slide 3:

Before we get to our analysis, I want to briefly introduce one important technique called StarSpace. It is a Natural Language Processing technique developed by Facebook to deal with text classification. The main idea is very simple. We have different labels for different documents. If we represent labels and documents using dense vector and embed them in the same space, documents with similar labels should be close to each other and also their corresponding labels as well. The plot in the right shows how StarSpace works in a high level. We use cosine similarity to calculate the distance between two different objects.

slide 4:

If you remember from the previous slide, there are many modes in StarSpace.

Here we extend this idea to embed co-assay data in the same space. We use TagSpace, mode 0 in StarSpace, to train our model. We think mode 0 is the most appropriate one, since we have two different kinds of objects, kmers and genes, for one cell label.

Similar to what is shown above for word, the plot below contains kmers, genes, and cells in the same space. Some genes and kmers are closes to certain cells, whereas others may be close to more than one cell.

slide 5:

I am sure most of you are very familiar with ngram. Here I just want to do a short recap. In NLP, words construct a sentence. ngrams of different length may construct a sense-group. In addition, they may occur together for many times in different documents. For example, the word “this” may often be followed by the word “is”. We can also apply the same idea to kmers and gene based on the idea of co-occurrence. Kmers are very similar to the concepts of words in NLP, so we train them treat as ngrams. Besides, genes with similar express level in the cells can be grouped together when the model trains them using ngram.

slide 6:

This is the loss function for StarSpace, which uses k negative sampling. I am not going to explain in detail. The main benefit is that it updates only weights associate with the positive sample and k negative samples instead of the weights for the whole NN, which increases the efficiency. So this is all the background about NLP technique I want to discuss with you.

slide 7, 8, 9:

This is the procedure for the whole project. (see slide)

The first part is to process dna data. The code CellSpace is mostly written by Zakieh, and I am sure you are all very familiar with her work already. (see slide) The programs read these barcodes, counts matrix, peaks, and reference sequences, generating result like what will be shown in the next slide. The kmers are generated with normal distribution. Those are closer to the peak have a higher probability to appear in the result file.

The second part is to process rna data. We load the data as Seurat objects, and normalize this data. We sample genes for cells based on the counts and assign weights. If these genes are more expressed in certain cells, the weights are higher. In addition, the weights of genes are sorted in ascending orders, which can be helpful when training StarSpace using ngram.

In addition to that, I merged both rna data and dna data into another file. In total, we have three files to feed to StarSpace, and we can compare the results for these three different files later.

slide 10:

These are the parameters I used to train StarSpace file.

slide 11, 12:

Let’s first look at the ad data. This is the original plot in Prof Chen’s paper. He did some data cleaning and the plot looks nice. This slide contains the plot when I tried to reproduce his work. I discussed this difference with Prof Chen. He and I agreed that it is extremely difficult to reproduce his work in that paper exactly due to the package updated for cisTopic. Despite the difference, he thought that my reproduction tsne plot reserves the overall pattern. On the contrary, I changed only his code from tsne to umap and produce the plot on the right. In this plot, we can barely see any pattern.

slide 13, 14, 15:

These are 2d visualization of cell embedding. TSNE looks better because it cares only about the local structure. So even though UMAP looks ugly, I will focus on UMAP later. Through comparison of different embedding size, it seems that the higher embedding size (within a reasonable range) can improve the result. It is not very obvious through comparison of size equals 15 and 20, because the true cell type labels are seriously imbalanced, the main difference may be revealed by cluster of minor class. Thus, I did not put a plot here. However, there is a downside for increasing embedding dimension: this will consume much memory. I am having some memory problem running with high dimension data in my laptop. Thus, the following plot for ad data is embedding size of 15, unless I specified.

Through eyeballing, we think that the co-embedding does look better than the embedding for rna or dna independently. The cells of same cell type stay close in combined data compared to rna or dna trained independently. The improvement compared to the rna is more obvious. We can see in this plot, the cluster 2 is separated into two sub groups. In addition, the distance between different cluster is not too large, meaning different classes are not that separable compared to co-embedding.

In this plot, we can see class label 2-11 are grouped together. They are excitatory neuron types. class label 13-16 are grouped together. They are inhibitory neuron types. class label 19-21 are grouped together. They are oligodendrocyte types. Others are non-neuron cells.

Thus, I think embedding captured the overall structure between different kinds of cells which is manifested through UMAP here.

slide 16:

We also ran Phenograph to cluster the cells. Apparently, it only identified major classes. Phenograph is superior to Kmeans and DBSCAN in our data. The latter two algorithms barely identified more than 5 clusters. Zakieh proposed that maybe Phenograph is not the best clustering algorithm here, and that we could try Seurat clustering with higher resolution. I met some technical problems doing so, but it seemed that the result is not improved significantly with resolution equals 1.3.

slide 17:

I will talk briefly about AMI, which is a measurement of clustering result.

slide 18-20:

The percentage is calculated based on the true cell type label, meaning that the sum of each row is equal to 100%. From the AMI score of Phenograph result, we prove from another side that the result for combined data is superior to dna or rna data alone.

slide 21-22:

Besides cell embedding, we tried to explore the relationship between genes and cells. Here is a cosine similarity heatmap between cluster biomarkers and cells. We filtered the genes by adjusted p value <= 0.01, select at most 10 per cluster. The redder the color, the closer of the embedding in the space. The plot is originally genes\*cells. Nevertheless, the plot is too long so I transpose it. Now, each row is a cell, and each column represent a gene. They are colored according to their class label. We can see that the distance between cluster and biomarkers are close indeed. Here is a cluster version.

slide 23:

Now, we focus on the neonatal data, namely p0 data. We are now looking at plot in Prof.Chen’s paper. Overall, it looks good, a little bit noisy in the dna data.

slide 24:

Here is plot for StarSpace embedding. It seems that the model does not learn anything from the RNA data. I am not quite sure what went wrong here. But the embedding for dna data and combined data is excellent, even better compared to the adult data umap plot. Clusters are easily distinguishable.

You may wonder why the label is not continuous for different types. For ad data, I sorted them based on their names. But for p0 data, the true labels provided by Prof.Chen contain class id already. Thus, I did not change them.

Excitatory neurons: 0, 1, 2, 5, 6, 7

Intermediate progenitor cells (IP): 3, 10, 13

Migrating inhibitory neurons (IN): 8, 9

Astrocytes/radial glia (Ast/RG): 4

Cajal-Retzius(CR): 14

slide 25:

Phenograph.

slide 26-28:

Here is the confusion matrix for phenograph. For RNA data, the AMI score is really bad. Since different classes are mixed. Besides, the combined data have roughly the same AMI as the dna data, since RNA data provides little extra information.

slide 29:

Zakieh might have told you a great deal about Palantir package, and the following pseudo-time analysis are completed using this package. Honestly, I am not familiar with this, so I directly using the true label from Prof Chen’s paper. The subset of p0 cell data includes only 5 classes. We can see very clear the only trajectory. Besides, the tsne for rna and dna is very similar, which is good news.

slide 30, 31:

Plot diffusion components for rna and dna.

slide 32, 33:

Corresponding Palantir result.