Predicting Cell Type from Spatial Transcriptomic Data

Ayushi Tandel
Author
atandel@stanford.edu

Nathaniel Chien
Author
nchien2@stanford.edu

Jordi Abante
Mentor
jabante@stanford.edu

Abstract

Spatial transcriptomics is up and coming field that focuses on where gene activity occurs in cell and how this affects cellular processes. Our project explores the question of whether spatial transcriptomic data can be used to identify a cell. Current methods for cell type identification primarily use gene expression data. We developed and tested a variety of models using spatial transcriptomic data, such as information about where genes localize in cell and cell morphology, to see if they compare to cell type predictors that only use gene expression data. In total, we implemented 5 deep learning models: 1) a baseline Multi-Layer Perceptron(MLP) that predicts cell type from gene expression data, 2) a Multi-Layer Perceptron(MLP) that predicts cell type from periphery scores, which are a measure of where a gene localizes in a cell, 3) a Convolutional Neural Network(CNN) that predicts cell type from images of cell morphology, 4) a Late-Fusion Multimodal Model(LFMM) that combines the outputs from models 2 and 3 to predict cell type from both periphery scores and cell morphology, and 5) a Early-Fusion Transformer(EFT) model that uses both periphery scores and cell morphology to predict cell type. Of these models, the EFT was most comparable to the baseline MLP model that predicts cell type from gene expression data. Though the EFT did not perform significantly better than the baseline gene expression model, the high accuracy of the model in spite of the relative increase in complexity shows that spatial transcriptomic data has potential to reveal more information about cell type.

1 Introduction

In 2020, Nature Methods crowned spatially resolved transcriptomics as the Method of the Year. Spatial transcriptomics is a cellular profiling method that scientists utilize to not only measure the amounts of gene activity in a cell but also localize the activity to sub-cellular components [1]. Spatial transcriptomics expands on prior gene expression research that mainly focused on RNA synthesis via RNA polymerases and transcription factors. In particular, these new methods seek to characterize how RNA localization controls gene expression [2].

Thus far, the majority of research on RNA localization is centered around the brain. The size and morphology of neurons–specifically their highly polarized environment, the long distances between the somas where RNA is synthesized, and the projections that occur in the axons–makes them model candidates for RNA localization studies. This research has also linked a number of neurological diseases, such as amyotrophic lateral sclerosis (ALS) and fragile X syndrome (FXS), to misregulated RNA localization in neurons [2]. These findings have sparked more curiosities and questions in the spatial transcriptomics field. Do similar cell types use a common set of mechanisms to ensure proper RNA localization? How important is RNA localization to gene function? Can we use data on where RNA localizes in a cell to identify its own cell type?

We implemented five deep learning models to tackle the question of whether data other than gene expression can be used to predict cell type. First, we implemented a baseline Multi-Layer Perceptron (MLP) model that predicts cell type based on gene expression data. Then, we implemented four

other models trained on different types of spatial transcriptomic data, which are extracted from the MERFISH dataset of cell images from the mouse primary motor cortex. Specifically, we used the cell boundary coordinates, which define the shape of different cell types, and periphery scores, which is a value between -1 and 1 for each gene in each cell based on how close it is to the cell boundary relative to the other genes in the cell. We developed a multi-layer perceptron (MLP) for the periphery scores and a Convolutional Neural Network (CNN) for cell boundary coordinates. After implementing separate models with these two types of spatial transcriptomic data, we explored late and early fusion methods to combine the multimodal data extracted from the MERFISH dataset. Compared to the baseline model's accuracy of 83.7%, our Late-Fusion Multimodal Model (LFMM) achieved an accuracy of 53% and our Early-Fusion Transformer (EFT) model achieved an accuracy of 83.54%. However, given the imbalanced nature of the MERFISH dataset, balanced accuracies are a better measure of classification ability across classes. The balanced accuracy of the Early Transformer model is 53.8% while the balanced accuracy of the baseline model is 47.8%. Though the EFT, our most comparable model to the baseline, did not perform significantly better than the baseline gene expression model, the high accuracy of the model in spite of the relative increase in complexity shows that spatial transcriptomic data has potential to reveal more information about cell type. These are exciting findings, especially since there has not been research in the spatial transcriptomics field proving that cell's spatial information can help distinguish it from other cells.

2 Related Work

With the growing availability of robust and accessible technologies, single-cell RNA sequencing (scRNAseq) has moved to the forefront of approaches for understanding cells at the molecular level [3]. Thus far, scRNAseq has proven to be a powerful method for identifying cell types and cell clusters based on gene expression profiles [4]. Since labelling scRNAseq data with cell types can be a manual, and time intensive process, a number of cell-type classifiers have been developed for scRNAseq data.

CellNet and SingleCellNet were among the first supervised classifiers for cell type prediction and use Random Forest methods to provide a similarity score between each cell type and each cell in a dataset [5, 6]. More recently, neural networks have become a popular method for cell-type annotation, due to the capacity of perceptron-based models to learn non-linear relations between classes and features [7]. LAmbDa, also known as label ambiguous domain adaptation, trained its neural network on raw data from multiple datasets and sought to correct batch effects by creating representations of labels shared across all datasets [8]. Another neural network model, SuperCT, devised a neural network that did 'online learning.' It not only trained on all datasets in the Mouse-Cell Atlas(MCA) but also added provided users an option to update the reference model by submitting new datasets online [9]. A highly accurate and fast neural network for cell-type classification is ACTINN, Automated Cell Type Identification using Neural Networks. Like other neural networks, this model trained on data from multiple sources and was proven to be robust against batch effects resulting from different sequencing methods [10].

Despite the popularity of single-cell RNA sequencing and the prevalence of cell-type classifiers for this data, current sequencing methods do not preserve knowledge about the spatial organization and connectivity of cells. Since sequencing methods involve cell disassociation, a vast amount of information about a cell's shape, size, and localization is discarded when gathering scRNA sequencing data. In response to this, researchers in the field are exploring cell imaging and processing techniques, that could allow a greater variety of features, including gene expression, gene localization, size, and shape, to be extracted from a cell [11]. Though the field of spatial transcriptomics is new, researchers are running into the same issues of producing data faster than they can manually label it.

A very small number of methods have been published for the automatic cell-type annotation of cell images and an even smaller number are compatible with the MERFISH dataset. JSTA, joint cell segmentation and cell type annotation for spatial transcriptomics, is one of the rare instances that was tested on MERFISH data. JSTA learns three distinct layers of information: a segmentation map, a pixel level classifier, and a cell level classifier. With these three layers, JSTA jointly segments cells in images and classifies them into granular cell subtypes. Though JSTA's algorithms are extendable to other spatial transcriptomics data, it relies on prior knowledge of cell-type specific gene expression [12]. Thus, it can only classify cell types with known and researched gene expression profiles. This is a serious drawback to JSTA, as it cannot achieve peak performance without accurate, detailed gene

Figure 1: Example Z-Slice of Cell

expression profiles for all cell types, and it has an inherent bias towards cell types that have been more widely researched. Within the spatial transcriptomics field and even more specifically for the MERFISH dataset, there are no published methods where cell annotation is the primary task without using a priori knowledge, revealing a gap in spatial transcriptomics research that our project hopes to fill.

3 Data

We will be using the MERFISH, multiplexed error-robust fluorescence in situ hybridization, dataset which is a high resolution spatial and projection map of cell types in the mouse primary motor cortex. The dataset of 27,000 cells and 252 genes is publicly available at: ftp://download.brainimagelibrary.org:8811/02/26/02265ddb0dae51de/. The dataset contains the raw MERFISH images from 2 mice, the processed images, segmented cell boundary coordinates, RNA spots for each gene expressed in a cell, and a cluster label assignment for each cell. We used the subset of cells that were labelled, leaving us with 18,000 cells to work with [11]. Each cell has a various number of z-slice images which show the cell boundary the RNA spots within it. An example of one z-slice of a cell, with RNA spots colored by gene, is shown in Figure 1.

3.1 Baseline Data

Our baseline model is a multi-class classifier for cell type based on gene expression. We counted up RNA spots for each gene in each cell to create a cell by gene matrix for our baseline model. We then normalized the gene expression in each cell according to the cell's overall expression and multiplied each resulting value by a scale factor of 10,000. We added 1 to all counts in the matrix, before calculating the log2 values of the matrix. Lastly, we filtered out outlier cells by removing cells with the highest and lowest 1% of overall expression as well as the cells with the highest and lowest 1% of standard deviation. The resulting matrix was split into the training, validation, and test sets for our baseline model using a 0.7, 0.15, 0.15 split.

3.2 Cell Morphology CNN Data

For the cell type predictor from cell morphology images, we used the segmented cell boundary coordinates extracted from the processed MERFISH images. Each cell had a series of (x,y) coordinates, which we plotted using matplotlib. We then converted these images to numpy arrays to use as image representations of the cell boundaries. These numpy arrays each have dimension (224, 224). Though each cell has multiple associated z-slices(views), we used only one slice–the median–for each image to avoid sample imbalances due to varying numbers of z-slices across cells. In order to be able to use this data as inputs to a CNN, we duplicated these arrays to simulate having 3 channels. The images were split into training, validation, and test with a 0.7, 0.15, 0.15 split.

Calculation of per-gene per-cell effect size

Let m be the count of spots for our gene of interest in a cell with n total spots.
The expected median rank is $med_{exp} = (n+1)/2$
The observed median rank med_{obs} is calculated from the spot ranks described above
Define the effect size per gene per cell as:

$$x = \frac{med_{exp} - med_{obs}}{med_{exp} - 1}$$

Since $1 <= med_{obs} <= n$, the effect size x is bounded from -1 to 1.
When $med_{obs} = 1$, then $x = 1$

$$x = \frac{\frac{n+1}{2} - 1}{\frac{n+1}{2} - 1} = 1$$

When $med_{obs} = n$, then $x = -1$

$$x = \frac{\frac{n+1}{2} - n}{\frac{n+1}{2} - 1} = \frac{\frac{n+1-2n}{2}}{\frac{n+1-2}{2}} = \frac{\frac{1-n}{2}}{\frac{n-1}{2}} = \frac{1-n}{n-1} = -1$$

When $med_{obs} = med_{exp}$, $x = 0$

Figure 2: Process for Calculating Periphery Scores

3.3 Periphery Score MLP Data

Our other model uses periphery scores calculated from the MERFISH data, which were provided by our external mentors. For each gene in each cell, the periphery score is a value between -1 and 1 measuring how peripheral or central a gene is within a cell relative to all the other genes that are expressed in that cell. Figure 2 outlines the process of calculating periphery scores in detail. A gene with a periphery score of 1 would be localized at the boundary of a cell while a gene with a periphery score of -1 would be localized at the center of a cell. Periphery scores thus contain information about both gene expression and localization. We created a matrix where the rows were unique cells (corresponding to images for MERFISH) and the columns were periphery scores for each gene in the dataset. The resulting matrix was split into the training, validation, and test sets for our periphery score MLP model using a 0.7, 0.15, 0.15 split.

3.4 Late Fusion Multi-Modal Model Data

The Late Fusion Multi-Modal Model uses periphery scores to train the MLP component and image boundary coordinates to train the CNN component. The pre-processing specifications are the same as the sections above.

3.5 Early Fusion Transformer Data

For our transformer, we encode each cell as a list of tokens, where a token represents a single RNA expression spot. Each token has 3 attributes, an integer representing the gene that's being expressed, the X-coordinate distance from the center of the cell, and the Y-coordinate distance from the center of the cell. Each list of tokens is padded to be equal to the length of the cell with the most tokens, in order to pass in a uniform input. With this representation, the information from both the periphery scores and the gene expression matrix has been implicitly encoded into the data. Like in the above models, the data was split into training, validation, and test sets with a 0.7, 0.15, 0.15 split.

4 Approach

4.1 Multi-Layer Perceptron

We developed an MLP for our baseline model and periphery score portion of our multi-modal model using the model specifications and parameters provided by ACTINN, Automated Cell Type Identification using Neural Networks. The ACTINN model has been tested to be fast and accurate at predicting cell type on a number of published gene expression datasets, which is why we chose to follow this model as our baseline [10]. Figure 3 shows the model architecture of an input layer, three hidden layers and an output layer. The model uses three hidden layers with 100, 50, and 25 nodes, respectively and uses the RELU activation function for the three hidden layers and the softmax activation function for the output layer [10]. The model is versatile and can work with any vectorized

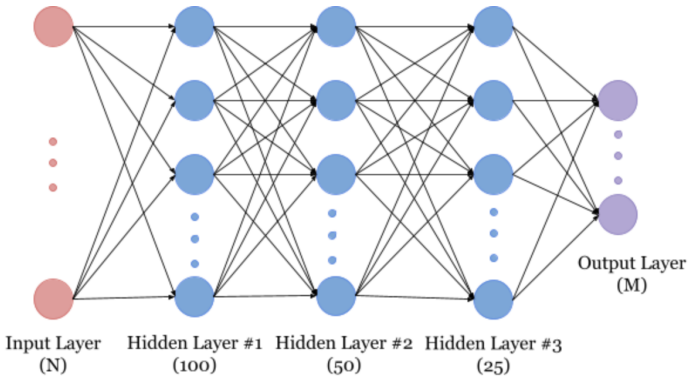

Figure 3: Multi-layer Perceptron Architecture

data by making sure the input(N) and output(M) sizes match the data type. For the baseline model, N was 252 and M was 93. For the periphery score model, N was 192 and M was 80.

We also followed many of the hyperparameter choices of the ACTINN model, including a starting learning rate of 0.0001 with staircase exponential decay, where the decay rate is to 0.95 and the decay step is 1000. Every 1000 global steps, the learning rate is multiplied by 0.95. We trained the baseline model and the periphery score model with a mini batch size of 128, choosing a number of epochs that balanced training and validation accuracy through manual tuning. The baseline model performed consistently well after 50 epochs of training, while the periphery score model performed consistently well after 200 epochs of training. [10].

4.2 Cell Morphology CNN

Since the cell boundary coordinates are a sparse form of data and the ResNet model we implemented for the Project Milestone did not perform well on the cell boundary images, we tried a different approach using a simple CNN. The CNN architecture comprises of a 3x3 2D Convolutional layer with 32 units, a 2x2 2D MaxPool layer, a Flatten layer, a Dropout layer with 0.9 probability, a Dense layer with 64 units and relu activation, and final Dense output layer with 80 nodes and softmax activation. We manually optimized the model by finetuning hyperparameters. The model consistently performed best with the Adam optimizer, a learning rate of 0.01, and 10 epochs of training.

4.3 Late Fusion Multi-Modal Model(LFMM)

The Late Fusion Multi-Modal Model takes the outputs of the Periphery Score MLP and the Cell Boundary CNN. Both models output a vector of shape (N, 80), where N refers to the number of samples and 80 refers to the number of possible cell type classifications. Using a for loop structure, the model determines a weight between 0 and 1 to multiply each model's output by, and the two weights (one for each model) are constrained to equal 1. Though the weights fluctuated across runs, we typically saw the late fusion model give a weight greater than 0.9 to the periphery score model and a weight less than 0.1 to the cell boundary model.

4.4 Early Fusion Transformer(EFT)

The early fusion transformer aims to better represent the data in a way that maintains gene expression and spatial transcriptomics data more explicitly. It takes tokenized cells as input, where each cell is represented by a list of RNA spots, and each RNA spot is represented by it's coordinates relative to the center of the cell and it's gene type. Our model was based off of the model in the Keras code examples [13]. The model embeds these tokens into 32-dimensional inputs, which are passed into 2 transformer encoder blocks, which include residual connections, layer normalization, and dropout. The transformer blocks use 3 attention heads, one for each of the input attributes. The transformer

Model	Accuracy	Balanced Accuracy
Baseline	83.67%	47.8%
Periphery Score MLP	54.6%	18.5%
Cell Boundary CNN	11.3%	1.28%
LFMM	53.3%	22.4%
EFT	83.4%	53.2%

Figure 4: Results: Model Accuracies and Balanced Accuracies

encoder layers feed into a classification multi-layer perceptron heads, which outputs the cell type vectors.

5 Experiments

5.1 Model Development

Aside from the models described above, we also experimented with using a pre-trained ResNet to aid with cell-boundary classification. However, we determined that the ResNet was not particularly effective on the simple cell-boundary image, so we moved instead to a convolutional neural network.

There was also experimentation within each model, in tuning hyperparameters and testing different data preprocessing methods. One particularly interesting example was with pruning away underrepresented classes. In order to try to improve our baseline model, we experimented with removing any cell types that contained less than 50 samples in the dataset. This did significantly increase model accuracy, but we eventually decided to leave the dataset unpruned. The issue of imbalanced and insufficient data is an interesting problem, one worth looking further into counteracting.

5.2 Results

We used sklearn's accuracy and balanced accuracy metrics to evaluate our models. The metrics assess the model's overall accuracy and the adjusted accuracy with all classes weighted equally to address any data imbalances. Figure 5 summarizes and compares the results of all the models.

Our baseline gene expression model was able to achieve a test accuracy of 83.67%, which is about what is expected. Some papers report much higher accuracy, but the MERFISH dataset is somewhat imbalanced, with some cell types only having 1 example, and over half of the cell types having less than 100 examples. When the data was pruned to remove these underrepresented classes, the baseline accuracy increased to about 93%. However, we ultimately decided to leave the entire dataset intact in order to provide a more interesting challenge and to be able to classify all cells.

The periphery score classifier was able to achieve a test accuracy of 54%, which shows that there is a learnable connection between cell type and gene expression patterns. The cell boundary classifier achieved a fairly poor performance, with a peak accuracy of just over 11%. The task of predicting cell type from morphology is a difficult one, so low accuracy is to be expected. Even after finetuning hyperparameters, this accuracy did not improve much.

We also evaluated the balanced accuracy of the models, given that the MERFISH dataset does not have equal samples of cells across cell type. Our baseline gene expression model achieved a balanced accuracy of 47.8% while our periphery score model achieved a balanced accuracy of 18.5%. The cell boundary classifier's balanced accuracy was a dire 1.25%. Since there are 80 possible cell type classifications, the balanced accuracy of the cell boundary classifier revealed that the model was randomly assigning labels and no valuable information was learned.

The multimodal models showed more promising results. The late fusion model achieved a balanced accuracy of 22.4% which was interesting since this was higher than the balanced accuracy of the periphery score model alone, which was 18.5%. Though the cell boundary classifier did not seem to

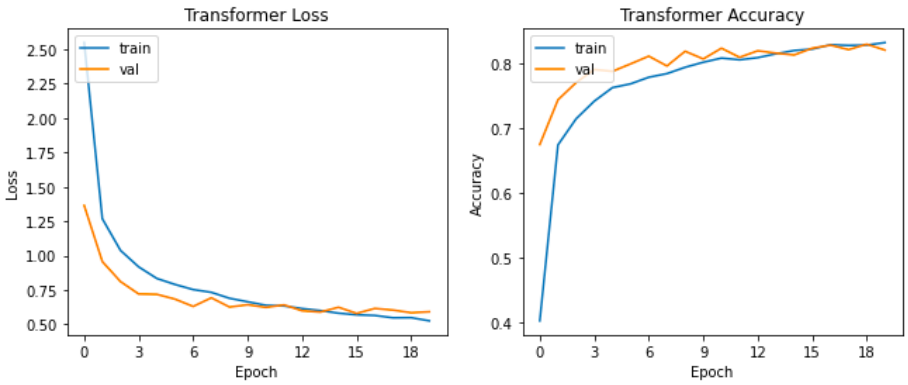

Figure 5: Results: Loss and accuracy curve for transformer model

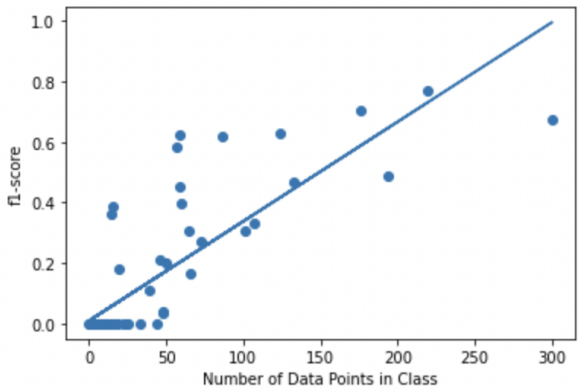

Figure 6: Positive Relationship Between Class Representation in Dataset and Class F1-Score

learn any valuable information on its own, combining its outputs with the outputs of the periphery score model increased the balanced accuracy. Our most promising results, however, came from the Early-Fusion Transformer model which achieved a an accuracy of 83.4% and a balanced accuracy of 53.2%. The Transformer's accuracy was comparable to the baseline's accuracy of 83.6% and the balanced accuracy was 5% higher than that of the baseline. Though the increase is modest, these results are exciting and suggest that including spatial transcriptomic data in cell type classification can help cell type classifiers increase balanced accuracy.

5.3 Further Analysis

The difference between the accuracy and balanced accuracy for all of these models show that the data is deeply imbalanced. To explore this further, we created a graph (Figure 6) for the Late-Fusion Multimodal Model mapping out the number of data points for each cell type class against the class's f-1 score, which is a measure of the classification ability. We used sklearn's classification_report metric to calculate each class's precision, recall, and f1-score. Though there are some exceptions, we generally saw that cell type classes with more data points achieved higher accuracy and balanced accuracy while underrepresented cell types in the dataset had very low accuracies. Specifically, cell type classes with less than 50 samples typically had f1-scores close to or of zero, while cell type classes with more than 50 samples had higher f1-scores. Since more that half of the cell types in our dataset are underrepresented, this graph explains the low overall accuracies and balanced accuracies. However, this also shows promise for future research, as we see good classification results when there is enough data for a cell type class, namely when there are 50 are more samples for each cell type in the dataset.

Our results for the early fusion transformer were the most promising for the models we tested. The accuracy did not differ significantly from that of the baseline, but the comparable results in spite of increased complexity lead us to believe that there is relevant information about cell type encoded within spacial transcriptomics data. The higher balanced accuracy of the transformer model may also indicate that this spatial data is a better indicator of cell type when given fewer data points.

6 Conclusion

The field of spatial transcriptomics is still in its infancy, and there is yet to be any extensive research on the relevance of RNA localization for predicting features such as cell type. Through experimentation with multi-layer perceptrons, convolutional neural networks, and transformers, we have succeeded in creating a model utilizing spatial transcriptomics data that has similar results to baseline models using gene expression data. The similar accuracy in spite of a more difficult optimization task gives us hope that there is indeed a learnable correlation between spatial data and cell type, and that there will be applications for this correlation in the future.

There is still plenty of future work to be done, from collection of more balanced data, to further analysis of results. We have seen from our analysis of class-specific accuracy that the imbalance on our dataset has caused poor predictive results for a subset of cell-types that are underrepresented in the dataset. Collecting more data for these cell-types, perhaps from other assays within MERFISH, could potentially improve results. Additional hyperparameter tuning could also potentially improve results of the transformer, since there wasn't sufficient time to find all the optimal parameters. Experimentation with dropout rates, learning rates, normalization techniques, and even the number of transformer blocks in the model could potentially improve results.

It's also worth delving deeper into our results, and determining exactly how much information is being encoded by spatial data as opposed to gene expression data. It would be interesting to determine if spacial data is particularly important for any specific cell types.

7 Code and Data Availability

All models were implemented in Jupyter Notebook and are available at https://github.com/nchien2/biods-final. Data is publicly available for download at ftp://download.brainimagelibrary.org:8811/02/26/02265ddb0dae51de/.

8 Contributions

Both authors collaborated with Sanket and Jordi Abante to brainstorm approaches for this project. Ayushi implemented the MLP model, CNN model, and the Late Fusion Multi-Modal Model. Nathaniel implemented a ResNet model for cell morphology for the Project Milestone and an Early Fusion Transformer model. Both authors shared ideas for the write-up with Ayushi taking the lead on the Abstract, Introduction, and Data sections and Nathaniel taking the lead on the Approach, Methods, and Conclusion section.

References

[1] Vivien Marx. Method of the year: Spatially resolved transcriptomics, Jan 2021.

[2] Krysta L. Engel, Ankita Arora, Raeann Goering, Hei-Yong G. Lo, and J. Matthew Taliaferro. Mechanisms and consequences of subcellular rna localization across diverse cell types. *Traffic*, 21(6):404–418, Apr 2020.

[3] Giovanni Pasquini, Jesus Eduardo Rojo Arias, Patrick Schäfer, and Volker Busskamp. Automated methods for cell type annotation on scrna-seq data, Jan 2021.

[4] Poulin JF;Tasic B;Hjerling-Leffler J;Trimarchi JM;Awatramani R;. Disentangling neural cell diversity using single-cell transcriptomics, Aug 2016.

[5] Patrick Cahan, Hu Li, Samantha A. Morris, Edroaldo Lummertz da Rocha, George Q. Daley, and James J. Collins. Cellnet: Network biology applied to stem cell engineering. *Cell*, 158(4):903–915, 2014.

[6] Yuqi Tan and Patrick Cahan. Singlecellnet: A computational tool to classify single cell rna-seq data across platforms and across species. *Cell Systems*, 9(2):207–213.e2, 2019.

[7] Michael Wainberg, Daniele Merico, Andrew Delong, and Brendan J Frey. Deep learning in biomedicine, Oct 2018.

[8] Travis S Johnson, Tongxin Wang, Zhi Huang, Christina Y Yu, Yi Wu, Yatong Han, Yan Zhang, Kun Huang, and Jie Zhang. LAmbDA: label ambiguous domain adaptation dataset integration reduces batch effects and improves subtype detection. *Bioinformatics*, 35(22):4696–4706, 04 2019.

[9] Peng Xie, Mingxuan Gao, Chunming Wang, Jianfei Zhang, Pawan Noel, Chaoyong Yang, Daniel Von Hoff, Haiyong Han, Michael Q Zhang, and Wei Lin. SuperCT: a supervised-learning framework for enhanced characterization of single-cell transcriptomic profiles. *Nucleic Acids Research*, 47(8):e48–e48, 02 2019.

[10] Feiyang Ma and Matteo Pellegrini. ACTINN: automated identification of cell types in single cell RNA sequencing. *Bioinformatics*, 36(2):533–538, 07 2019.

[11] Meng Zhang, Stephen W. Eichhorn, Brian Zingg, Zizhen Yao, Hongkui Zeng, Hongwei Dong, and Xiaowei Zhuang. Molecular, spatial and projection diversity of neurons in primary motor cortex revealed by in situ single-cell transcriptomics. *bioRxiv*, 2020.

[12] Russell Littman, Zachary Hemminger, Robert Foreman, Douglas Arneson, Guanglin Zhang, Fernando Gómez-Pinilla, Xia Yang, and Roy Wollman. Jsta: Joint cell segmentation and cell type annotation for spatial transcriptomics, Jan 2020.

[13] Ntakouris;. Timeseries classification with a transformer model, Jun 2021.