ATTENTION BASED MODELS FOR CELL TYPE CLASSIFICATION ON SINGLE-CELL RNA-SEQ DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Cell type classification serves as one of the most fundamental analyses in bioinformatics. It helps discovering new cell types, recognizing tumor cells in cancer microenvironment and facilitating the downstream tasks such as trajectory inference. Single-cell RNA-sequencing (scRNA-seq) technology can profile the whole transcriptome of different cells, thus providing invaluable data for cell type classification. Existing cell type classification methods can be mainly categorized into statistical models and neural network models. The statistical models either make hypotheses on the gene expression distribution which may not be consistent with the real data, or heavily rely on prior knowledge such as marker genes for specific cell types. By contrast, the neural networks are more robust and flexible, while it is hard to interpret the biological meanings hidden behind a mass of model parameters. Recently, the attention mechanism has been widely applied in diverse fields due to the good interpretability of the attention weights. In this paper, we examine the effectiveness and interpretability of the attention mechanism by proposing two novel models for the cell type classification task. The first model classifies cells by a capsule attention network (CAN) that performs attention on the capsule features extracted for cells. To align the features with genes, the second model first factorizes the scRNA-seq matrix to obtain the representation vectors for all genes and cells, and then performs the attention operation on the cell and gene vectors. We name it Cell-Gene Representation Attention network(CGRAN). Experiments show that our attention-based models achieve higher accuracy in cell type classification compared to existing methods on diverse datasets. Moreover, the key genes picked by their high attention scores in different cell types perfectly match with the acknowledged marker genes.

1 INTRODUCTION

Single-cell RNA-sequencing (scRNA-seq) is a transcriptomic analysis at single-cell resolution, which was first introduced by (Tang et al., 2009). Different from traditional bulk sequencing technology that produces the average gene expression values in mixtures of cells, scRNA-seq can measure the expression values of all genes in every single cell. This brings unprecedented opportunities for cell type classification, which analyzes the heterogeneity and diversity of cells based on their expression profiles (Liang et al., 2014; Muraro et al., 2016; Baron et al., 2016). Cell type classification is not only itself an important analysis to identify various cell types, such as recognize tumor or normal cells in cancer microenvironment and discriminate different cell states in cell differentiation, but also facilitates many downstream tasks such as cell trajectory analysis (Saelens et al., 2019) and cell-cell communication (Kumar et al., 2018). As many subsequent analyses highly depend on accurate identification of cell types, classification of single cells plays a key role in understanding diseases and biological processes.

Despite the progress of sequencing technologies and the boom of analytical methods, cell type classification still faces a lot of serious challenges. On the one hand, most existing statistical methods utilize the prior knowledge of ’marker genes’. The ’marker genes’ are such genes that have distinguishable expression levels in a particular cell type. Therefore, these methods annotate a cell with the cell type whose marker genes match well with the cell’s highly expressed genes. However, such markers-based manual annotation is time-consuming and irreproducible (Abdelaal et al., 2019). On the other hand, the quality of the scRNA-seq data always suffer from noises and dropouts of high percentage, up to 95% for instance (Huang et al., 2018; Pierson & Yau, 2015), which may seriously reduce the classification accuracy. To tackle these issues, some methods make assumption on the statistical distribution of the gene expression values, but these assumptions sometimes lead to a worse classification results and whether these assumptions capture the true distribution of the data awaits further discussion.

Unlike the methods relying on prior knowledge and assumptions, the machine learning models, especially the neural networks, can automatically extract useful cell features from the sparse and noisy data to classify cells, which avoids the negative effect of missing marker genes or misleading assumptions. A model that can both achieve high accuracy in cell type classification and provide vivid biological insights from its learned parameters is in urgent demand. In this paper, we examine both the effectiveness and interpretablity of the Attention mechanism. First applied in translating word sequences in the area of Natural Language Processing (NLP) (Vaswani et al., 2017; Devlin et al., 2018), the Attention mechanism has achieved great success in many other data types ranging from graphs to images (Velickovic et al., 2017; Dosovitskiy et al., 2020). For every input token, the attention layer renews the token embedding by a weighted aggregation on the embeddings of all tokens, where the attention weights are dynamically optimized and can reveal the close relationships among the input tokens. In the task of cell type classification, by viewing genes and cells as tokens, the attention weights among them may provide new biological knowledge.

We design two novel models in order to examine the performance and interpretability of the attention mechanism in the task of cell type classification. Our first model is Capsule Attention Network (CAN). Given a cell vector containing all genes' expression values in this cell, the network first uses fully connected layers (FC) to extract different capsule features from the cell vector. Then an attention layer is used to renew these features and capture the cell-type-specific patterns, followed by a convolutional layer to get the cell type classification result. Our second model is Cell-Gene Representation Attention Network (CGRAN) which learns the attention directly based on the representation vectors of genes and cells. It first obtains the gene vectors and cell vectors by factorizing the scRNA-seq matrix. For every cell to be classified, we input a token sequence with the first one being the cell vector and the following ones being the gene vectors to the attention layers. Local attention is adopted to learn a more variant and differentiate attention weights among genes from the same group. Finally, the renewed cell vector output by the attention layers is used for cell type classification. It is found that for every cell from a specific cell type, the genes picked by their large attention weights to the cell match well with the marker genes of the cell type. Besides, the attention weights among genes can divide the genes into groups that matches with known gene sets biologically defined. These intriguing explanation of the attention weights ensures strong model interpretability.

The main contributions of our article are as follows. (1) The self-attention mechanism applied in CAN learns cell-type-specific patterns from the representation capsules of the cell, which achieves a performance gain over existing methods. (2) By aligning the input sequence tokens with true biological entities, CGRAN learns the attention weights among the genes and the cell to be classified, which enables direct biological interpretation from the model's attention weights. (3) The attention masking technique utilized in CGRAN facilitates local attention operations on genes, which improves classification accuracy. (4) Experiments show that both CAN and CGRAN has strong transferability across different datasets having common genes and cell types.

2 Related Work

Existing methods for cell type classification can be mainly categorized into statistical models and neural network models. Most of the statistical models either heavily rely on prior knowledge such as marker genes for specific cell types, or make hypotheses on the gene expression.

Typical solutions for cell type classification are unsupervised clustering followed by annotation of clusters. CellAssign (Zhang et al., 2019a) develops a probabilistic model while Garnett (Pliner et al., 2019) designs a hierarchical model to utilize cell-type marker genes. Both methods are based on the same principle, that is, supervised clustering with prior knowledge (Lee & Hemberg, 2019). ScType developed a computational platform based on scRNA-seq data along with a comprehensive cell marker database as background information(Ianevski et al., 2022). Although leveraging cell-type markers can enhance the biological interpretability, all these methods highly rely on prior knowledge of markers, which may fail when lack of markers.

As neural networks have shown stronger ability in diverse areas, many methods deal with the noisy and high-dimensional single-cell RNA-seq data using neural networks. ACTINN (Ma & Pellegrini, 2020) employs a neural network with three hidden layers to predict cell type. EpiAnno (Chen et al., 2022) uses a Bayesian neural network to embed the cells into a latent space, where the cells follow a Gaussian mixture distribution. Cell BLAST (Cao et al., 2020) projects the cells in high-dimensional transcriptomic space to a low-dimensional cell embedding space, and then search for similar cells. OnClass (Wang et al., 2021) embeds cell types into a low-dimensional space, maps each cell into the region of its cell type and classifies cells into different cell types in Cell Ontology. scCapsNet (Wang et al., 2020) designs a deep-learning architecture of capsule networks and analyzes the internal weights parameters among capsules, while it is hard to effectively interpret the biological meanings hidden in the parameters. By contrast, the parameters in our model CGRAN have strong links with biological knowledge, which guarantees the interpretability of our model.

3 Problem Statement

Cell type classification problem is to assign each cell to a correct cell type from k different known cell types $\{l_1, l_2, ..., l_k\}$. Formally, Given a sparse scRNA-seq matrix $M^{c \times g}$ depicting the expression values of g genes in c cells, with c_l cells having their ground-truth cell types, the method needs to predict the cell types for the other $c - c_l$ cells.

4 Methods

In this section, we elaborate on two attention based models, namely the Capsule Attention Network (CAN) and Cell-Gene Representation Attention Network (CGRAN) for cell type classification on scRNA-seq data.

4.1 Capsule Attention Network

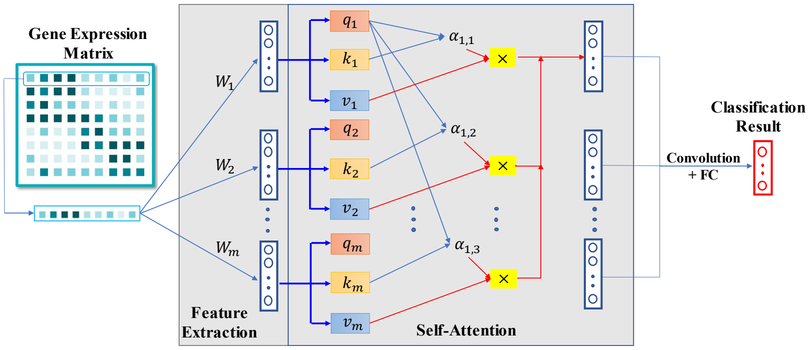


Figure 1: Capsule Attention Network

The framework of our capsule attention network is shown in Figure 1. The first part is feature extraction. We use capsule vectors to capture different features of the cell to be classified. A capsule is a set of neurons or a vector that can represent characteristic of a cell. We use u Fully-Connected (FC) layers followed by ReLU activation and L2-normalization to extract u capsule features $C_{i,1}, C_{i,2}, ..., C_{i,u}$ for the i-th cell from its cell vector. Formally,

$$C_{i,n} = \text{L2Norm}(\text{ReLU}(W_n M_{i,:} + b_n)) \tag{1}$$

where W_n and b_n are the parameters of the n-th FC layer extractor and $M_{i,:}$ is the vector of the i-th cell. The second part is a single-head self-attention layer. Self-attention mechanism is used to learn latent relationships among different capsules and update their representations, denoted as $C^{'}_{i,j}$. We also add L2-normalization on these new representations.

$$(C^{'}_{i,1}, C^{'}_{i,2}, ...C^{'}_{i,u}) = \text{L2Norm}(\text{Attention}(C_{i,1}, C_{i,2}, ...C_{i,u})) \tag{2}$$

At last, we input attention representations to a convolution layer followed by a FC layer, which finally outputs the possibilities of a cell belonging to different cell types.

4.2 Cell-Gene Representation Attention Network

Although the attention weights learned by CAN help to renew the latent cell features extracted from different aspects and may finally contribute to a more accurate cell type classification result, the close relationships among cells and genes still remain uninterpreted. Here we present our second model that has strong biological interpretability, namely Cell-Gene Representation Attention Network. The architecture of our model is presented in Figure 2.

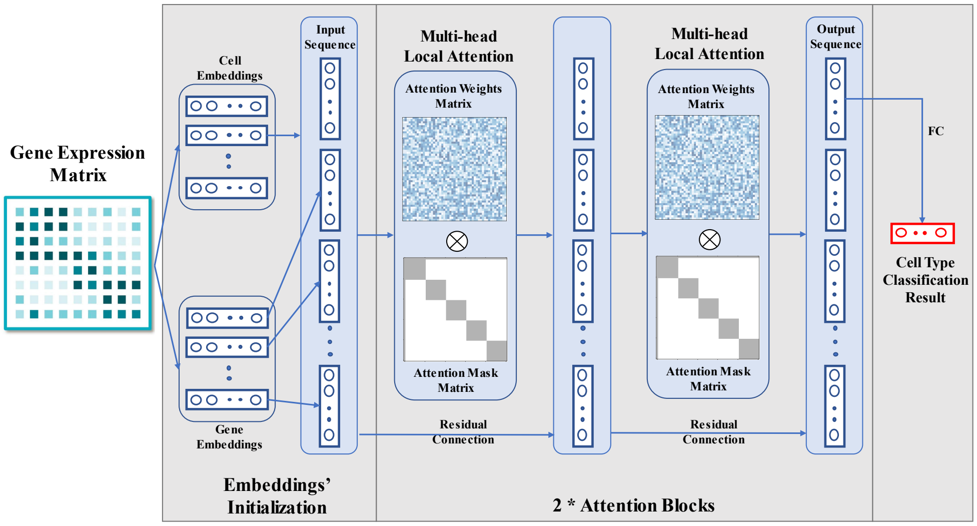


Figure 2: Cell-Gene Representation Attention Network

The model first obtains the d-dimentional representation vectors of cells $C^{c\times d}$ and genes $G^{g\times d}$ by factorizing the scRNA-seq matrix M. Two different ways of matrix factorization have been evaluated. The first way is Singular Value Decomposition (SVD), which factorizes the input scRNA-seq matrix as:

$$M = U\Sigma V^T \tag{3}$$

where $U^{c\times c}$ and $V^{g\times g}$ are real orthogonal matrices, and $\Sigma^{c\times g}$ is a rectangular diagonal matrix with r non-negative real numbers in descending order on the diagonal, with r being the rank of M. Here we denote Σ_s as the rectangular diagonal matrix of size $c \times g$ with the diagonal values being the square roots of the values in Σ. We obtain cell SVD vectors as $U\Sigma_s$ and gene SVD vectors as $V{\Sigma_s}^T$. We choose the first d dimensions as the representation vector for each gene and cell since they correspond to d biggest singular values. Therefore, for cell i, $C_{i,:} = (U\Sigma_s)_{i,1:d}$ and for gene j, $G_{j,:} = (V{\Sigma_s}^T)_{j,1:d}$.

The second way of matrix factorization is to first randomly initialize the cell embedding vectors $C^{c\times d}$ and the gene embedding vectors $G^{g\times d}$, and optimize them by Mean Square Error (MSE) loss:

$$\arg\min_{C,G} \text{MSE}(M, CG^T). \tag{4}$$

After obtaining the cell vectors and gene vectors, given cell i to be classified, we generate a sequence $\boldsymbol{S}_i$ of vectors with the first one being the cell vector, and the following g ones being the sum of the cell vector and the corresponding gene vectors. Formally,

$$\boldsymbol{S}_{i,0} = C_i, \quad \boldsymbol{S}_{i,j} = C_i + G_j, j \in \{1, 2, ..., g\} \tag{5}$$

The second part of CGRAN model consists of two sequential attention blocks. Each block consists of a multi-head attention layer, a residual connection using fully connected network and L2-normalization on the outputs. However, the problem arises as using fully attention in a long sequence will lead to small attention weights, which will further decrease classification accuracy. In other words, the attention weights matrix, denoted by W, will be filled with values very close to zero, leading to the attention weights undistinguishable from each other. As a result, attention mechanism may fail to learn the relationships among embeddings.

To solve the problem, local attention mechanim is introduced. Genes are divided into different groups. Only the attention weights among genes from the same group are preserved, whereas those from different groups are neglected. Two different methods are proposed for grouping. The first one is a simple but effective one. We treat every hundred genes as a group. The method is named as uniform grouping. The second one groups the genes by clustering and is named as gene cluster grouping. For implementation, we use K-Means to cluster genes into different groups. The advantage of the first method is that it is a simple and easy way for implementation. Also, every group has the same number of genes, which makes attention weights comparable with each other. By contrast, the second grouping method is based on the similarities of gene expressions. However, this makes gene group imbalanced, as the number of genes in different groups can be of big difference.

For local attention implementation, we conduct element-wise multiplication on attention weights matrix with attention mask matrix, denoted by F. The multiplication result is denoted by LW(local attention weights matrix). Attention mask matrix is a $(g+1)*(g+1)$ matrix, consisting of only 0 and 1. The first row and column of the attention mask matrix are filled with 1s because these are attention weights among cells and genes. On positions for gene pair from the same group, there will be 1 in attention mask matrix, 0 otherwise. Then L2-normalization($Norm$) will be implemented on the local attention weights matrix(LW), which serves as the Softmax function in the original attention mechanism. The following formulas describe the implementation of local attention.

$$\begin{aligned} &F_{0,j} = F_{0,j}, j \in \{0,1,2,...,g\} \\ &F_{ij} = \begin{cases} 1, gene\ i\ and\ gene\ j\ in\ same\ group \\ 0, else \end{cases} \end{aligned} \tag{6}$$

$$\begin{aligned} &W = Softmax(\frac{QK^T}{\sqrt{d}}) \\ &LW_{i,j} = W_{i,j} * F_{i,j} \\ &LW = Norm(LW) \end{aligned} \tag{7}$$

In the last part of CGRAN, the first token's embedding of the sequence will be passed into a fully-connected layer to generate the possibilities of a cell belonging to different cell types.

5 Experiments

5.1 single-cell RNA-seq datasets

We elaborate the nine datasets used for experiments, listing out their numbers of cells, genes, cell types. The datasets we select include cells from different species, such as human and mouse and have a wide range of cell numbers from few hundreds to nine thousands.

Table 1: Single-Cell RNA-Seq Datasets' Descriptions

Dataset	CRC	GSE70580	GSE72056	GSE75688	GSE96993	NSCLC	PBMC	Spleen human	Spleen mouse
Cell Number	8496	647	4636	515	334	9051	5356	4406	4432
Gene Number	12547	26087	22280	27420	10827	12415	14218	14064	12699
Cell Type Number	20	4	7	5	4	16	5	7	7

5.2 Data Preprocessing

For every cell in the data, we first divide every gene's expression level by the sum of gene expression levels in the cell. This makes genes' expression levels among cells comparable. Then, we multiply the gene expression levels with a constant 100000 and employ a log transform to adjust the expression level in the data into a reasonable range. After that, we calculate the variances of genes and then pick out the top 1000 largest among them. Highly variable genes are more likely to help us distinguish different cells as they provide more information on cell heterogeneity. Also, there are tens of thousands genes in human and mouse genome. Choosing the top 1000 largest variance genes also serves as dimension reduction of high dimensional scRNA-seq data.

5.3 Parameter Setting

For CAN, we generate 16 capsule vectors. Each of the capsule has a dimension of 128. We set attention layer's output embedding dimension to 16. The convolution layer's kernel size is 5 and output channel is 8. During the training process, cross entropy loss is used as the loss function. The activation function in CAN is ReLU. A dropout probability of 0.2 is used on all layers. Adam optimizer is used and learning rate is set to 0.0001 with training epochs set to 50.

For initialization of cell embeddings and gene embeddings in CGRAN, we use the whole matrix after data preprocessing as the input of matrix factorization for both initialization methods and set the embedding dimension of cell embeddings and gene embeddings to 128. For NN-based Matrix Factorization, dropout is set to 0.1 and learning rate is set to 0.005 and training epoch is set to 130. In both attention blocks, we set the head number of local attention layer to 10. In the first attention block, the output embedding's dimension for each head is 8. Then we concatenate every head's outputs together, which are the inputs for the second attention block. In the second attention block, the output embeddings' dimension for each head is 4, and we concatenate every head's outputs together. As a result, the second attention block's output embeddings' dimension is 40. Cross entropy loss is used as the loss function. A dropout probability of 0.1 is used on all layers. Adam optimizer is used and learning rate is set to 0.0001 with training epochs set to 75.

For cross validation on both CAN and CGRAN, we split every dataset into training set and test set with the ratio of 4:1 randomly. For fair evaluation, our methods and other baselines are repeatedly run three times to get the average accuracy. More detailed experimental settings are presented in Appendix A.1.

5.4 Experimental Result

We first compare CAN and CGRAN with previous methods, including Support Vector Machine(SVM), Random Forest(RF), scCapsNet, ACTINN, Cell Blast, scVI, Moana and XGBoost. We do not compare with previous methods using prior knowledge or domain knowledge, for example, Garnett uses marker genes as prior knowledge, OnClass uses cell ontology as domain knowledge. From Table 2, CAN and CGRAN has a robust performance with high accuracy on all datasets. For the column of 'CGRAN', we select the best performance of CGRAN model among all of its settings for each dataset. For SVM, we adopt scikit-learn implementation. In fact, scikit-learn's SVM implementation adopts "one-versus-one" approach for multi-class classification. As a result, SVM's classification accuracy might be high due to the integration on all sub-models. The top three methods on each dataset are shown in bold.

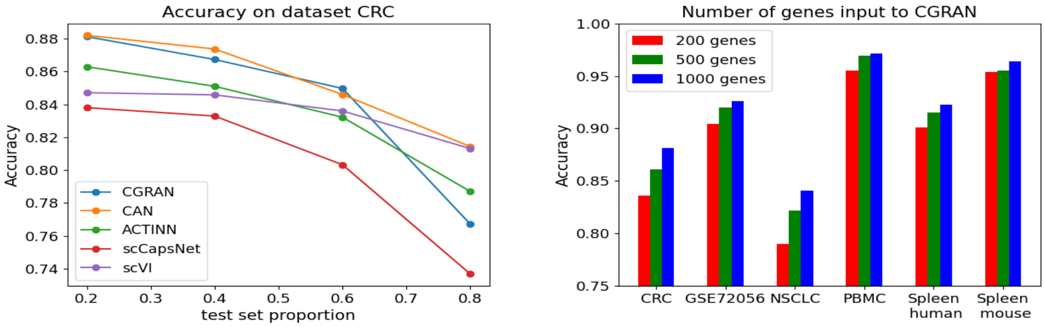
Table 2: Accuracy comparison with previous methods

	CAN	CGRAN	SVM	RF	scCapsNet	ACTINN	Cell Blast	scVI	Moana	XGBoost
CRC	**88.20%**	**88.12%**	**89.64%**	81.47%	83.80%	86.29%	68.79%	84.71%	45.29%	85.47%
GSE70580	**96.92%**	**97.30%**	**96.92%**	94.15%	96.15%	96.15%	95.52%	91.54%	93.84%	96.15%
GSE72056	**93.75%**	**92.78%**	92.34%	91.59%	92.21%	92.56%	87.62%	91.59%	78.44%	**93.53%**
GSE75688	**94.17%**	**93.20%**	92.23%	92.23%	90.77%	91.26%	79.61%	92.23%	91.26%	**93.20%**
GSE96993	**82.83%**	80.59%	**82.08%**	**82.08%**	77.61%	79.10%	70.96%	80.60%	56.71%	79.10%
NSCLC	**83.26%**	**84.10%**	**83.26%**	79.01%	79.14%	82.72%	69.14%	**83.99%**	34.67%	83.05%
PBMC	**97.94%**	97.39%	97.57%	**98.00%**	**97.94%**	97.85%	91.86%	97.57%	**97.94%**	97.76%
Spleen_human	**91.49%**	**92.29%**	91.26%	87.64%	90.28%	91.04%	87.20%	89.23%	39.45%	**91.72%**
Spleen_mouse	**96.73%**	96.39%	**97.29%**	92.33%	95.38%	**96.73%**	91.54%	95.26%	95.60%	96.28%

CGRAN's performances under different model's architecture settings are listed in table 3. From the table, we can see that SVD with uniform grouping and local attention, NN-based MF with uniform grouping and local attention have better and robust performance among different architecture's settings of CGRAN. Compared with using fully attention, CGRAN models with local attention have better performances.

Table 3: Accuracy comparison under different CGRAN's settings

Dataset	NN-based MF + fully attention	NN-based MF + uniform grouping + local attention	NN-based MF + gene cluster grouping + local attention	SVD + uniform grouping + local attention
CRC	77.58%	85.52%	84.88%	**88.12%**
GSE70580	95.38%	**97.30%**	96.92%	96.15%
GSE72056	88.68%	**92.78%**	92.45%	92.62%
GSE75688	86.40%	**93.20%**	**93.20%**	**93.20%**
GSE96993	73.13%	**80.59%**	79.10%	77.61%
NSCLC	75.15%	81.15%	79.95%	**84.10%**
PBMC	97.35%	**97.39%**	97.29%	97.13%
Spleen human	89.79%	91.38%	91.72%	**92.29%**
Spleen mouse	88.38%	95.15%	93.79%	**96.39%**



(a) (b)

Figure 3: (a) Classification accuracy with different test set proportion on CRC dataset. (b) Classification accuracy with different number of genes

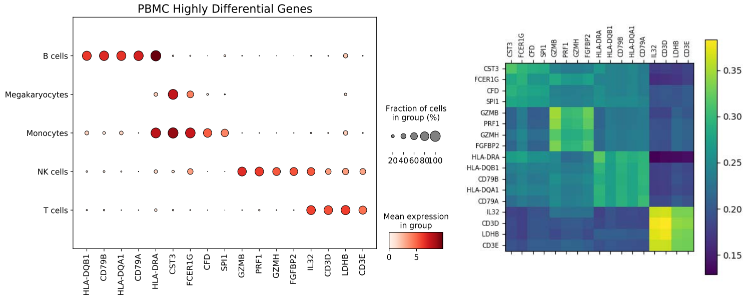
To further test different models' robustness and dependency on the scale of training data, we split datasets according to different training set and test set ratios and evaluate models' performance based on test set accuracy. As shown in Figure 3(a), both CAN and CGRAN perform consistently well on different test set proportion of CRC. More results on other datasets are illustrated in Appendix A.3.

Moreover, the number of genes input into the CGRAN can also affect its performance. The input sequence length of CGRAN equals to the number of highly variable genes + 1. The more genes are selected, the more cells and genes' information are fed into the models. Due to computational cost and devices' limitations, we here provide experiments on CGRAN with gene number varied from 200 to 500 to 1000. From Figure 3(b), as gene number becomes larger, classification accuracy becomes higher. In this paper, we use top 1000 most variable genes for cell type classification. If computational feasible, we expect CGRAN will achieve higher classification accuracy with large number of genes.

5.5 The Interpretation of Cell-Gene Representation Attention Network

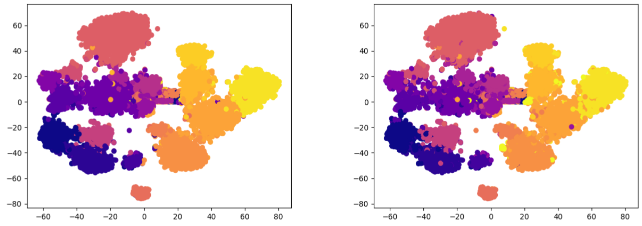
In this part, we interpret our model using attention weights from two aspects. We first analyze the attention weights among cells and genes, which provides a method for identification of marker genes. For each cell, we select those genes with top-k largest attention weights into a "top-k list". Then, if a gene appears in the 'top-k lists' of high proportion cells of a certain cell type, it is called high attention gene(HAG) for that cell type. Moreover, by discriminating high attention genes(HAG) among different cell types, we find out the unique ones for each cell type. We call them highly differential genes. The highly differential genes identified by CGRAN match well with the acknowledged marker genes provided by CellMarker (Zhang et al., 2019b) and Panglao database Franzén et al. (2019).

In practice, we set k to 50, and proportion to 0.7 and use SVD + uniform grouping and local attention model. Figure 4(a) offers the highly differential genes found in PBMC dataset. From HLA-DQB1 to HLA-DRA are B cells' highly differential genes. From CST3 to SPI1 are monotypes' highly differential genes. From GZMB to FPFBP2 are NK cells' highly differential genes. The rest are T cells' highly differential genes. Among them, CD79A,CD79B, GZMH, CD3D,CD3E are well-known marker genes. Figure 4(a) also shows these highly differential genes' distinguishable expression level in different cell types. The bigger and darker the circle is, the more differentially expressed the gene is in that cell type. By CGRAN, marker genes can be identified by analyses on cell-gene attention weights. More examples on other datasets can be found in AppendixA.5.1.



(a) (b)

Figure 4: (a) Highly differential Genes found in PBMC and their distinguishable expression levels. (b) The attention weights among all highly differential genes in PBMC.



(a) (b)

Figure 5: Visualizations of cell embeddings on CRC dataset, colored according to (a) the predicted cell types by CGRAN, (b) groundtruth cell type labels of CRC.

Then in the second aspect, we also illustrate the attention weights among all highly differential genes identified by CGRAN in PBMC. As Figure 4(b) shows, although many of the highly differential genes come from different groups, those differential genes of the same cell types have larger attention weights among each other and smaller attention weights with genes from different cell types. Moreover, from literature search, we find that these highly differential genes for specific cell type have similar functions and form a 'gene set'. For example, CD3D and CD3E are two highly differential genes for T cells. The proteins encoded by CD3D and CD3E are parts of the T-cell receptor/CD3 complex (TCR/CD3 complex) and are involved in T-cell development and signal transduction. Please refer to Appendix A.5.2 for more details.A.5.2.

Finally, Figure 4(c) offers a visualization of cell embeddings output by the second attention blocks after using TSNE for dimension reduction. Every dot in the figures corresponds to a cell in CRC dataset and are colored by the predicted cell types given by CGRAN(Figure 5(a)) or groundtruth cell types(Figure5(b)). The model can effectively learn classification-friendly embeddings. For visualization of cell embeddings on more datasets, please refer to the Appendix A.5.3.

5.6 Experiments on Transfer Learning across datasets

In this section, we demonstrate CGRAN's transferability across datasets. GSE72056 and PBMC datasets have genes and cell types in common, which makes the base for transfer learning. In Table 4, we illustrate cell types and the number of cells in different cell types in GSE72056 and PBMC. T cells , B cells and NK cells are three common cell types between two datasets while each dataset has its own distinct cell types. We calculate every gene's variance in PBMC and select the top 1000 most variable genes which appear in both datasets. CGRAN is pretrained on GSE72056 for a nine-cell-type classification task, including all cell types in GSE72056 and PBMC.

Table 4: GSE70256 and PBMC 's cell types

	Tumor cells	T cells	B cells	Macrop -hages	Endoth -elial	Cancer-associated -fibroblasts	NK cells	Monoc -ytes	Megakary -ocytes
GSE72056	1751	2066	515	126	65	61	52	0	0
PBMC	0	2660	696	0	0	0	583	1398	19

Then, we compare three transfer settings on PBMC dataset. The first one(setting A) directly takes the initialization of cell embeddings and gene embeddings using SVD as input into the attention blocks and only finetunes on the last fully connected layer. The second setting(setting B) adopts the same embeddings' initialization as the first setting but finetunes on the whole CGRAN model. The last one(setting C) employs a different method for embedding initialization. Gene embeddings on GSE72056 are directly transferred and used as gene embeddings on PBMC. Cell embeddings are initialized by gradient descent with the goal to minimize MSE loss mentioned in CGRAN method4. Then we finetune the whole CGRAN model.

Table 5: CGRAN's transferability from GSE72056 to PBMC

	setting A	setting B	setting C
GSE72056	92.13%	92.13%	92.13%
PBMC 80% for finetune 20% for test	19.59%(epoch 1) 83.40%(epoch 75)	29.66%(epoch = 1) 92.72%(epoch = 25) 97.39%(epoch = 75)	67.63%(epoch 1) 96.18%(epoch 25) 96.83%(epoch 75)

Table 5 shows big improvement of transferability when finetuning on the whole model. The finetuning process also converges faster as test accuracy exceeds 90% only using 25 epochs.
In the third transfer setting of CGRAN, the highly differential genes for GSE72056 with the pretrained CGRAN model match with the highly differential genes for PBMC with the finetuned CGRAN model. They both point out that IL32, CD3D, CD3E, CD2 and CD3G are highly differential genes for T cells, CD79A and CD79B are highly differential genes for B cells. These genes are well recognized marker genes for T cells and B cells. This not only indicates that CGRAN's has strong transferability across datasets, but also proves that CGRAN provides reasonable and stable interpretations for its architecture and helps to discover marker genes. Moreover, CGRAN has the ability to predict novel cell types during transfer learning. Further explorations on CAN and CGRAN's transferability are presented in Appendix A.6.

6 Conclusion

In this article, we propose two attention-based models for single-cell RNA-seq data cell type classification. Capsule Attention Network has high classification accuracy compared with previous models. To further illustrate cells and genes' relationships, we provide Cell-Gene Representation Attention. It also performs classification task well on different datasets. By comparing the attention weights among different genes and cells, we select the highly differential genes for different cell types, which match with the acknowledged marker genes. By visualizing attention weights, we discover that highly differential genes from the same cell type share closer relationships and interactions, which may be an indication of their similar gene functions.

REFERENCES

Tamim Abdelaal, Lieke Michielsen, Davy Cats, Dylan Hoogduin, Hailiang Mei, Marcel JT Reinders, and Ahmed Mahfouz. A comparison of automatic cell identification methods for single-cell rna sequencing data. *Genome biology*, 20(1):1–19, 2019.

Maayan Baron, Adrian Veres, Samuel L Wolock, Aubrey L Faust, Renaud Gaujoux, Amedeo Vetere, Jennifer Hyoje Ryu, Bridget K Wagner, Shai S Shen-Orr, Allon M Klein, et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. *Cell systems*, 3(4):346–360, 2016.

Zhi-Jie Cao, Lin Wei, Shen Lu, De-Chang Yang, and Ge Gao. Searching large-scale scrna-seq databases via unbiased cell embedding with cell blast. *Nature communications*, 11(1):1–13, 2020.

Xiaoyang Chen, Shengquan Chen, Shuang Song, Zijing Gao, Lin Hou, Xuegong Zhang, Hairong Lv, and Rui Jiang. Cell type annotation of single-cell chromatin accessibility data via supervised bayesian embedding. *Nature Machine Intelligence*, 4(2):116–126, 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Oscar Franzén, Li-Ming Gan, and Johan LM Björkegren. Panglaodb: a web server for exploration of mouse and human single-cell rna sequencing data. *Database*, 2019, 2019.

Mo Huang, Jingshu Wang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, Roberto Bonasio, John I Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. Saver: gene expression recovery for single-cell rna sequencing. *Nature methods*, 15(7):539–542, 2018.

Aleksandr Ianevski, Anil K Giri, and Tero Aittokallio. Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data. *Nature communications*, 13(1):1–10, 2022.

Manu P Kumar, Jinyan Du, Georgia Lagoudas, Yang Jiao, Andrew Sawyer, Daryl C Drummond, Douglas A Lauffenburger, and Andreas Raue. Analysis of single-cell rna-seq identifies cell-cell communication associated with tumor characteristics. *Cell reports*, 25(6):1458–1468, 2018.

Jimmy Tsz Hang Lee and Martin Hemberg. Supervised clustering for single-cell analysis. *Nature Methods*, 16(10):965–966, 2019.

Jialong Liang, Wanshi Cai, and Zhongsheng Sun. Single-cell sequencing technologies: current and future. *Journal of Genetics and Genomics*, 41(10):513–528, 2014.

Feiyang Ma and Matteo Pellegrini. Actinn: automated identification of cell types in single cell rna sequencing. *Bioinformatics*, 36(2):533–538, 2020.

Mauro J Muraro, Gitanjali Dharmadhikari, Dominic Grün, Nathalie Groen, Tim Dielen, Erik Jansen, Leon Van Gurp, Marten A Engelse, Francoise Carlotti, Eelco Jp De Koning, et al. A single-cell transcriptome atlas of the human pancreas. *Cell systems*, 3(4):385–394, 2016.

Emma Pierson and Christopher Yau. Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome biology*, 16(1):1–10, 2015.

Hannah A Pliner, Jay Shendure, and Cole Trapnell. Supervised classification enables rapid annotation of cell atlases. *Nature methods*, 16(10):983–986, 2019.

Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods. *Nature biotechnology*, 37(5):547–554, 2019.

Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382, 2009.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *stat*, 1050:20, 2017.

Lifei Wang, Rui Nie, Zeyang Yu, Ruyue Xin, Caihong Zheng, Zhang Zhang, Jiang Zhang, and Jun Cai. An interpretable deep-learning architecture of capsule networks for identifying cell-type gene expression programs from single-cell rna-sequencing data. *Nature Machine Intelligence*, 2 (11):693–703, 2020.

Sheng Wang, Angela Oliveira Pisco, Aaron McGeever, Maria Brbic, Marinka Zitnik, Spyros Darmanis, Jure Leskovec, Jim Karkanias, and Russ B Altman. Leveraging the cell ontology to classify unseen cell types. *Nature communications*, 12(1):1–11, 2021.

Allen W Zhang, Ciara O'Flanagan, Elizabeth A Chavez, Jamie LP Lim, Nicholas Ceglia, Andrew McPherson, Matt Wiens, Pascale Walters, Tim Chan, Brittany Hewitson, et al. Probabilistic cell-type assignment of single-cell rna-seq for tumor microenvironment profiling. *Nature methods*, 16 (10):1007–1015, 2019a.

Xinxin Zhang, Yujia Lan, Jinyuan Xu, Fei Quan, Erjie Zhao, Chunyu Deng, Tao Luo, Liwen Xu, Gaoming Liao, Min Yan, et al. Cellmarker: a manually curated resource of cell markers in human and mouse. *Nucleic acids research*, 47(D1):D721–D728, 2019b.

A Appendix

A.1 Detailed Experimental Settings

We choose $linear\_kernel$ and set $max\_iteration$ to 1000 for SVM using scikit-learn implementation.

For random forest and XGBoost algorithm, we also adopt scikit-learn implementation. Number of trees in the forest(namely $n\_estimators$) in random forest is set to 1000 and other parameters are set using the default configurations. For XGBoost, $n\_estimators$ is set to 100 and other settings follows the default configuration as well.

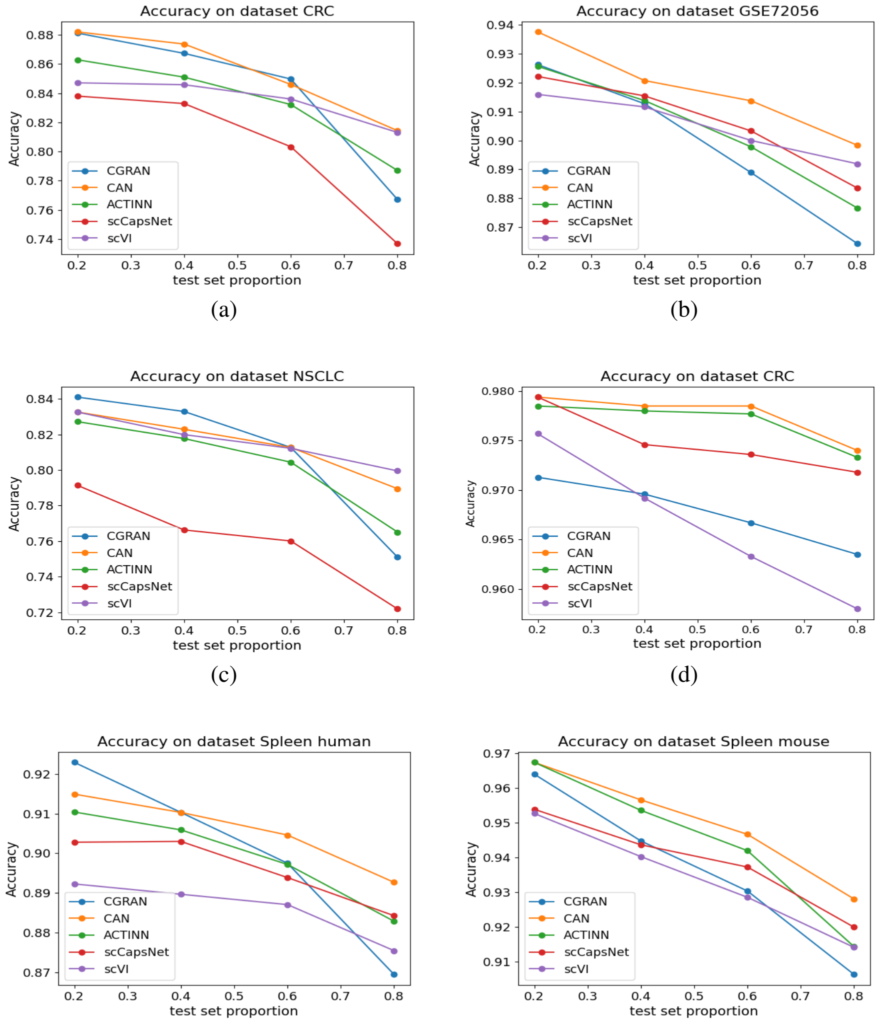
For ACTINN and scCapsNet, we follow their implementation instructions mentioned in their articles. For Cell Blast, we try different hyper-parameters for different datasets and and best results are preserved in Table 2 . For scVI, we use the implementation provided by its authors. $Hiddendimensionality$ is set to 256 and $latentdimensionality$ set to 128. The number of layer is set to 2 and dropout probability is set to 0.3. For Moana, we experiment on different hyper-parameters for different datasets and the best results are presented in Table 2.

A.2 Data Preprocessing

Before training the models, data preprocessing is implemented. For every cell in the data, we first divide every gene's expression level by the sum of gene expression levels in the cell. This makes genes' expression levels among cells comparable. Then, we multiply the gene expression levels with a constant 100000 and employ a log transform to adjust the expression level in the data into a reasonable range. After that, we calculate the variances of genes and then pick out the top 1000 largest among them. Highly variable genes are more likely to help us distinguish different cells as they provide more information on cell heterogeneity. Also, there are tens of thousands genes in human and mouse genome. Choosing the top 1000 largest variance genes also serves as dimension reduction of high dimensional scRNA-seq data.

A.3 DIFFERENT SPLITTING RATIO OF TRAINING AND TEST DATASETS

In this section, we present experiments on different ratio of training and test set on more datasets.



(a) (b)

(c) (d)

(e) (f)

Figure 6: Accuracy v.s. different train/test ratios on different datasets

From the figures above, we can see that both CAN and CGRAN perform consistently well on different test set proportion in all datasets. CGRAN's accuracy drops when the proportion of test set becomes larger because of its deep architecture. More data are needed for training appropriate cells and genes' embeddings as well as the attention weights among them. Also, as the proportion of test set becomes larger, the attention weights for cells from rare cell types are harder to learn.

A.4 More about model's interpretations

A.4.1 CAN's interpretations

We visualize both the attention weights among the different capsule features of a cell and the renewed features output by the attention layer in CAN. Three random cells are sampled from each cell type in dataset GSE70580. The attention weights and renewed features of every cell are visualized by heatmaps. For each row representing a cell type, the three figures in the left are the visualizations of the attention weights among features in the three cells, and the three figures in the right are visualizations of the output vectors of attention layer after concatenation of the three cells. The deeper the color is in the heatmap, the larger the value is in that position. As shown in Figure 7, cells from the same cell type share a similar pattern in attention weights and output vectors while cells from different cell types differ from each other.

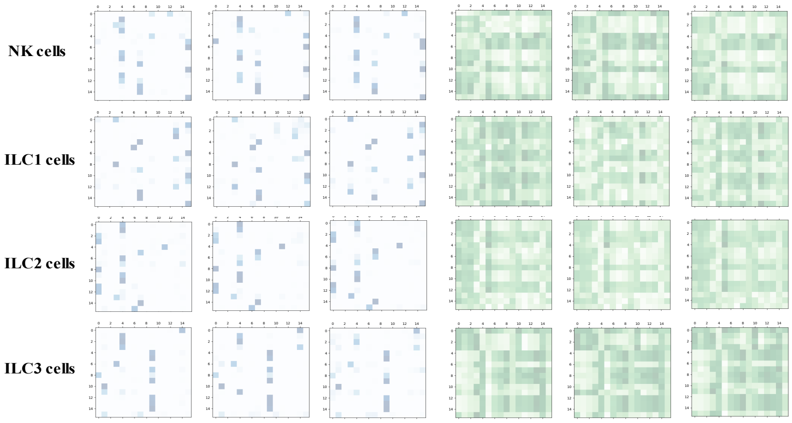


Figure 7: heatmaps of attention weights and output vectors of attention layer on GSE70580

A.5 CGRAN's interpreatations

A.5.1 Cell-Gene attention and discovery of marker genes

Here we illustrate the highly differential genes identified on GSE72056 and GSE70580 as well as their distinguishable expression levels among cell types.

On dataset GSE72056, CD79B to TCL1A are highly differential genes of B cells. FCGRT is highly differential genes for Endothelial. From FCER1G to RNF130 are highly differential genes for Macrophages. From IL32 to CD8B are highly differential genes for T cells.

On dataset GSE70580, ILC cells are cells that have close relationships with T cells. So some of marker genes of ILC cells may be the same as T cells. As the Figure 9 shows, CD3D is highly differential gene for ILC1 cell type. IL32 and GATA3 are the highly differential genes for ILC2. From AMICA1 to PCDH9 are ILC3 highly differential genes. The rest are NK cells' highly differential genes.

A.5.2 More about gene-gene attention weights of CGRAN

Here we provide more analyses on attention weights among genes, especially among highly differential genes. We first select highly differential genes and visualize their attention weights with the genes in their groups. To be specific, we visualize attention weights among highly differential genes for NK cells,B cells,T cells and the genes in the same group in PBMC dataset. It is found that highly differential genes have larger attention weights among each other while smaller attention weights with most of the other genes in their group. This indicates that these highly differential genes form a 'gene set'. Moreover, from literature search, these highly differential genes for specific cell type have similar functions. For example, CD3D and CD3E are two highly differential genes for T cells.

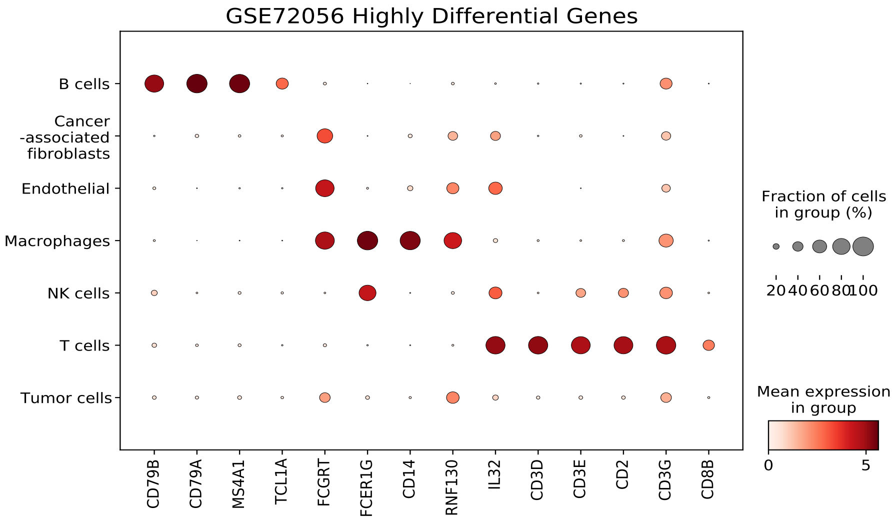


Figure 8: GSE72056 highly differential genes

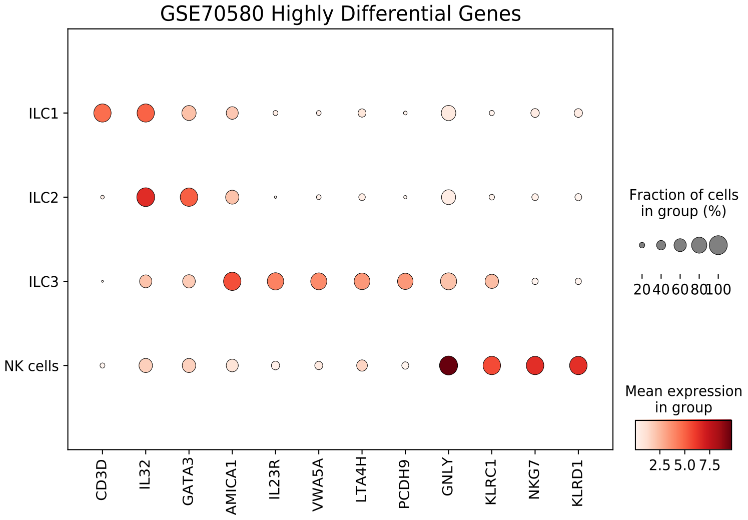


Figure 9: GSE70680 highly differential genes

The proteins encoded by CD3D and CD3E are parts of the T-cell receptor/CD3 complex (TCR/CD3 complex) and are involved in T-cell development and signal transduction.

A.5.3 Visualization on cell embeddings

Here more visualization results of the cell embeddings before the final fully connected layer in CGRAN are presented. Using TSNE to implement dimension reduction to two dimensionality on the cell embeddings, we can see from the Figure 5 and Figure 11 that cells from different cell types are separated from each other while cells from same cell types form clusters. This indicates CGRAN learns classification-friendly embeddings.

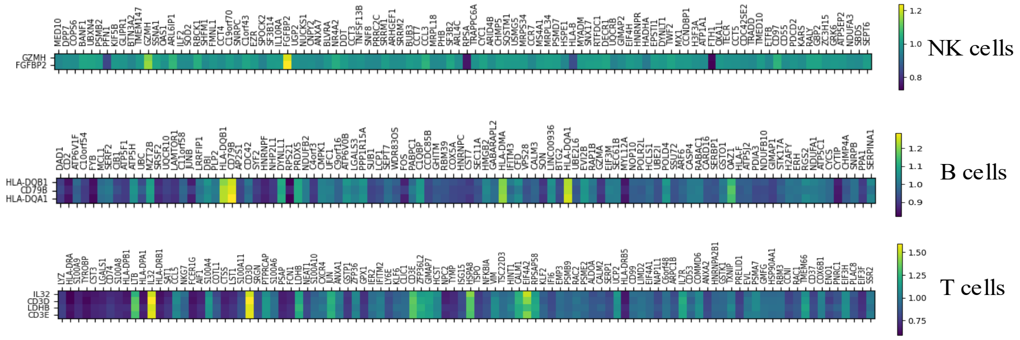
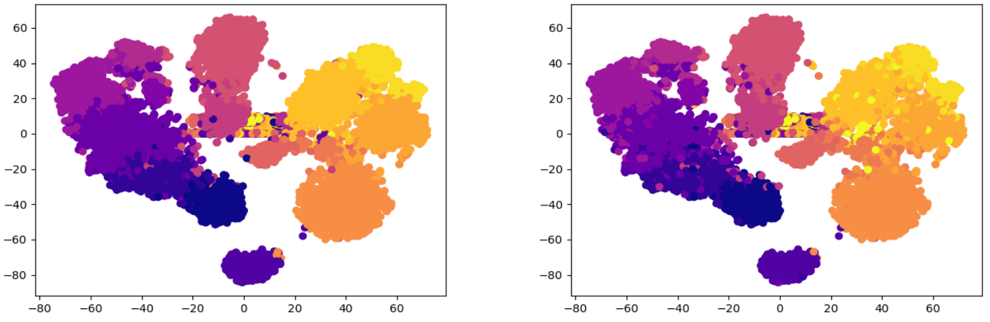


Figure 10: Highly differential genes attention weights among genes in same group of PBMC



(a)

(b)

Figure 11: NSCLC cell embeddings' tsne visualization. (a) Different colors represent different cell types predicted by CGRAN on NSCLC. (b) Different colors represent different cell types given by the groundtruth labels of NSCLC.

A.6 More Experiments on Transfer Learning across datasets

In this section, we demonstrate more experiments of CAN and CGRAN's transferability across GSE72056 and PBMC.

A.6.1 CAN's transferability

We first calculate each gene's variance in GSE72056 and select top 1000 most variable genes that also appear in PBMC dataset. These 1000 genes are used for both GSE72056 and PBMC dataset. Considering that there are both common cell types and distinct cell types between two datasets, we train three CAN models on GSE72056. For the first model, only three cell types are considered. They are T cells, B cell and NK cells. The second model performs a four-category classification task: T cells, B cells , NK cells and 'other cell types'. The last one performs a nine-category classification task, including all cell types in GSE72056 and PBMC. After training three models on GSE72056, we finetune them on PBMC dataset. We use 30% cells from PBMC dataset for finetuning and 70% for testing. During the finetuning process, only parameters in the last fully connected layer which generate classification result are being updated and other parameters trained on dataset GSE72056 remain unchanged.

The result in Table 6 illustrates classification accuracy on test set of PBMC finetuning for 1 epoch and 50 epochs(finish finetuning). We can discover that CAN has the ability to capture important

Table 6: CAN's transferability from GSE72056 to PBMC

	3 labels	4 labels	9 labels
GSE72056	98.67%	94.07%	92.02%
PBMC 30% for finetune 70% for test	87.38%(epoch 1) 90.72%(epoch 50)	87.36%(epoch 1) 89.68%(epoch 50)	61.31%(epoch 1) 89.20%(epoch 50)

features for cell type classification and transfer them to similar dataset. As the last column shows, CAN has the ability to predict novel cell types that are not present in the original dataset.

A.6.2 CGRAN'S TRANSFERABILITY

As CGRAN learns cells and genes' embeddings as well as their relationships, we also test CGRAN's transferability across datasets using the same 1000 genes, the same classification category(3 labels, 4 labels, 9 labels) as CAN. However, CGRAN requires the initialization of cell embeddings and gene embeddings using matrix factorization on each dataset. We implement SVD method for cell embeddings and gene embeddings initialization separately on dataset GSE72056 and PBMC. Same as CAN, only parameters from the last fully connected layer are finetuned.

Table 7: CGRAN's transferability from GSE72056 to PBMC(only finetune on the last FCN)

	3 labels	4 labels	9 labels
GSE72056	98.10%	94.07%	93.31%
PBMC 20% for finetune 80% for test	56.09%(epoch 1) 54.95%(epoch 75)	62.57%(epoch 1) 64.97%(epoch 75)	37.85%(epoch 1) 72.11%(epoch 75)
PBMC 80% for finetune 20% for test	62.87%(epoch 1) 72.20%(epoch 75)	62.87%(epoch 1) 72.20%(epoch 75)	38.53%(epoch 1) 77.89%(epoch 75)

From the above table, we can discover that CGRAN has the ability to transfer knowledge from one dataset to another. It also has the ability to predict novel cell types during transfer learning. With more data provided for finetuning, CGRAN will achieve better classification performance.

Then, we calculate every gene's variance in PBMC dataset and select the top 1000 most variable genes that appear in both PBMC and GSE72056. Classification accuracy of different gene selection choices are presented in Table below.

Table 8: CGRAN's transferability from GSE72056 to PBMC(Different gene choices)

	9 labels (GSE72056 top 1000 common genes)	9 labels (PBMC top 1000 common genes)
GSE72056	93.31%	92.13%
PBMC 80% for finetune 20% for test	38.53%(epoch 1) 77.89%(epoch 75)	19.59%(epoch 1) 83.40%(epoch 75)

Using different genes indeed has impact on transfer learning's accuracy on PBMC dataset. However, both gene choices come to high classification accuracy.