Peng *et al. BMC Bioinformatics* 2019, **20**(Suppl 8):284
https://doi.org/10.1186/s12859-019-2769-6

BMC Bioinformatics

RESEARCH Open Access

Combining gene ontology with deep neural networks to enhance the clustering of single cell RNA-Seq data

Jiajie Peng[1,2,3], Xiaoyu Wang[1] and Xuequn Shang[1,2*]

From International Conference on Data Science, Medicine and Bioinformatics
Wenzhou, China. 22- 24 June 2018

Abstract

Background: Single cell RNA sequencing (scRNA-seq) is applied to assay the individual transcriptomes of large numbers of cells. The gene expression at single-cell level provides an opportunity for better understanding of cell function and new discoveries in biomedical areas. To ensure that the single-cell based gene expression data are interpreted appropriately, it is crucial to develop new computational methods.

Results: In this article, we try to re-construct a neural network based on Gene Ontology (GO) for dimension reduction of scRNA-seq data. By integrating GO with both unsupervised and supervised models, two novel methods are proposed, named GOAE (Gene Ontology AutoEncoder) and GONN (Gene Ontology Neural Network) respectively.

Conclusions: The evaluation results show that the proposed models outperform some state-of-the-art dimensionality reduction approaches. Furthermore, incorporating with GO, we provide an opportunity to interpret the underlying biological mechanism behind the neural network-based model.

Keywords: Single cell RNA-seq data, Gene ontology, Autoencoder, Neural network

Background

In the past decade, transcriptome studies have benefited from next-generation sequencing (NGS) based on RNA expression profiling (RNA-seq) [1–3]. However, the resulting expression value based on bulk RNA-seq is an average of its expression levels across a large population of input cells [4]. Such bulk expression profiles are insufficient to provide insight into the stochastic nature of gene expression [5]. Therefore, bulk measures of gene expression may not help researchers to understand the distinct function and role of different cells [4]. To address the problem, single cell RNA-seq (scRNA-seq) is applied to assay the individual transcriptomes of large numbers of cells [6, 7]. The gene expression at single-cell level provides an opportunity for better understanding of cell function and new discoveries in biomedical areas [8, 9].

ScRNA-seq data analysis poses several new computational challenges. To ensure that the single-cell based gene expression data are interpreted appropriately, it is crucial to develop new computational methods. One of the most important applications of scRNA-seq is to group cells and identify new cell types. The major computational challenge in this application is to cluster cells based on the gene expression at single-cell level. Clustering based on scRNA-seq data may help us to understand underlying cellular mechanisms, which can promote the discovery of new markers on specific types of cells [10], and identification of tumor subtypes [11], etc.

In the clustering problem, cells are partitioned into different cell types based on their global transcriptome profiles. Each cell type has a significantly distinctive expression signature from the others. Since the

*Correspondence: shang@nwpu.edu.cn
[1]School of Computer Science, Northwestern Polytechnical University, 710072 Xi'an, China
[2]Key Laboratory of Big Data Storage and Management, Northwestern Polytechnical University, Ministry of Industry and Information Technology, 710072 Xi'an, China
Full list of author information is available at the end of the article

expression values are always with high dimensionality and noise from the sequencing result, dimensionality reduction is usually performed before clustering. Till now, several methods have been proposed to eliminate the influence of noise and reduce the dimension. The existing methods could be loosely grouped into two categories, unsupervised method and supervised method.

In the unsupervised category, the main idea is to perform dimensionality reduction before clustering. The simplest method is based on the principal component analysis(PCA) [12]. As one of the most popular methods for dimensionality reduction, PCA has been studied extensively for clustering single cells [13–16]. Assuming that the data is normally distributed, PCA uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables, which are called principal components. However, for scRNA-seq datasets, they are not exactly linearly separable. T-distributed stochastic neighbor embedding (t-SNE) [17] is a nonlinear dimensionality reduction technique, which is also used for clustering single cells recently [15, 16]. Based on the Gaussian kernel, t-SNE converts high dimension data to low dimension space. But, it usually maps multidimensional data to two or more dimensions suitable for human observation. Hence it always accompanies with dimension restriction. Besides, similar to PCA, t-SNE also does not consider the drop out events of scRNA-seq data. To consider the specificity of scRNA-seq data, ZIFA[18] uses zero-inflated factors to deal with the drop out events in scRNA-seq data. Assuming that drop out events may lead to zero counts, ZIFA models these counts exactly zero rather than close to zero in the dataset. The evaluation test shows that ZIFA performs better than PCA and t-SNE on some datasets. But the hypothesis of ZIFA is that zero is inflated as Gauss distribution, and the transformation between the descending dimension and original data is linear. Given the expression profiles of single cells, SNN-Cliq computes the similarity between cells by using the concept of shared nearest neighbor (SNN), and implements clustering algorithm based on graph theory [19]. By combining multiple clustering methods, SC3 performs a consensus clustering which includes spectral transformation, k-means algorithm, and complete link approach to achieve high accuracy and robustness [20]. However, SC3 and SNN-Cliq cannot build a relationship between data representation and quantity and property of cell types. Integrating PCA and hierarchical clustering, pcaReduce tries to improve the original PCA method by finding a connection between the PCA-based representations and the number of resolvable cell types. Meanwhile, denoising autoencoder (DAE) [15] is used to reconstruct the data from high dimensions to low dimension space.

Motivated by the success of neural networks in other areas, Lin et al. develop a supervised method to generate the low dimensional representation of scRNA-seq data based on neural networks (NN) [15]. NN model combines the neural network with the protein-protein interaction (PPI) network to classify a number of cells. Given cells with know cell types, this model can be trained as a supervised model. After that, the hidden layer of the trained neural networks is used for generating the low dimensional representation of scRNA-seq data. The experimental test shows that this supervised method performs better than most of the existing unsupervised models.

Although many attempts have been made to cluster single cells based on the global transcriptome profiles, most of them only consider the transcriptome profiles neglecting the prior biological knowledge. This large limits the performance of state-of-art systems. Inspired by the success of neural network in modeling the hierarchical structure and function of a cell [21], we ask whether combining the rich prior biological knowledge in gene ontology (GO) with neural networks could enhance the clustering of cells based on their global transcriptome profiles. Gene Ontology (GO) [22], which has been widely used in many areas [23–28], provides a popular vocabulary system for systematically describing the attributes of genes and other biological entities. As one of the most popular bioinformatics sources, it contains reliable and easy-interpreted prior biological knowledge. In this article, we try to construct the structure of neural networks based on the prior knowledge of GO. By integrating GO with both supervised and unsupervised models, two novel methods are proposed, named GOAE (Gene Ontology AutoEncoder) and GONN (Gene Ontology Neural Network) respectively, for clustering of scRNA-seq data. The major contributions of this article are as follows:

- To better dimensionality reduction of scRNA-seq data, we propose a novel neural work structure considering the prior knowledge in GO.
- We propose two novel models, named GOAE and GONN, to enhance cluster cells based on their transcriptome profiles.
- The evaluation results show that the proposed models outperform some state-of-the-art approaches.
- Incorporating with GO, we provide an opportunity to interpret the underlying biological mechanism behind the neural network-based model.

Methods

We propose a novel model to obtain the low dimensional representation of scRNA-seq data by combining the Gene Ontology and neural network model. We use the terms in GO to replace the neuron in the neural

network and convert the fully-connected neural network as partial-connected. Based on this idea, we propose two novel methods: an unsupervised method based on an autoencoder model and a supervised method based on a traditional neural network model. The basic idea of our models is to perform the dimensionality reduction by training a neural network (or autoencoder) model and extract the latent layer as low dimensional representation. This section consists of the following components. First, we will introduce how to select significant GO terms from the whole GO structure. Second, we combine GO with an autoencoder to build an unsupervised model for dimensionality reduction, named GOAE. Third, we combine GO with a neural network to build a supervised model for dimensionality reduction, named GONN. Finally, the low dimensional representation is used for clustering of cells based on a clustering method.

Selection of significant GO terms

Gene Ontology (GO) is a popular vocabulary system for systematically describing the attributes of gene and gene product. Each GO term could annotate a set of genes. GO consists of three different categories, which are biology process, molecular function and cellular component. GO is structured as a directed acyclic graph. Each term has defined relations with other terms in the same or various categories. In this step, we choose GO terms that are used in the following model. We only use terms in the biological process and molecular function category since these terms might be more functional related. In GO, a parent term annotates all the genes annotated by its descendants. The main steps of selecting GO terms used in the following steps are as follows.

First, we select all the GO terms in the third layer. Evaluation test shows that GO terms at the third layer can achieve the best performance. The number of GO terms at different levels is shown in Table 1. These 1543 GO terms at the third level are the candidate terms that connect with the input layer in the neural network.

Second, we remove redundancy terms from the candidate terms obtained from the last step. The annotated genes of different terms may have overlap. Therefore, we remove the redundancy terms to decrease the information redundancy and the parameters in the following neural network-based model.

Table 1 Number of Gene Ontology terms at different layers

layer number	0	1	2	3	4	5	6	7	8	9	10	11
biology process	1	24	151	1010	2662	3934	3443	2167	784	305	98	15
molecular function	1	17	111	476	894	1397	887	481	196	63	23	4

Specifically, let $GO_i : \{gene_1, gene_2, \cdots gene_n\}$ be a GO term, named GO_i, annotating a set of annotation genes $gene_1, gene_2, \ldots gene_n$. The unique score U_{ij} of two GO terms is defined as follows:

$$U_{ij} = \frac{num(GO_i \cap GO_j)}{num(GO_i \cup GO_j)}. \quad (1)$$

If the unique score U_{ij} of two GO terms is larger than 0.5, the two GO terms are considered as not unique. Then, we will delete the GO term that has fewer annotation genes.

Third, we remove the terms annotating genes that have similar expression profiles in different cells. Different genes may have different expression level in different cells. We tend to select the genes that have different expression levels for clustering. Therefore, we select the terms annotating genes having diverse expression levels in different cells. The diversity of a GO terms could be measured by gene expression values. Z-score-based method is used for normalization on gene dimension. Following this normalize operation, the expression values of each gene is normalized as a standard normal distribution. We define std_j as standard deviation of $gene_j$. The diversity score H_i of a GO term GO_i is calculated as follows:

$$H_i = \frac{\sum_{j=1}^{n} std_j}{n}, \quad (2)$$

where n is the number of genes annotated by GO_i. If the diversity score of GO_i is less than the given threshold (in this case 0.1), GO_i is considered as low diversity term. We then delete the low diversity GO terms.

After these three steps, we obtain a set of GO terms with low redundancy and high diversity.

Architecture of unsupervised model (GOAE)

In the task of scRNA-seq data clustering, an unsupervised dimensionality reduction model is a key component. To perform the dimensionality reduction, we combine the Gene Ontology with autoencoder that has been widely used in other areas, like image processing, natural language processing.

To combine the GO with neural network, we add GO terms to the neural network as partial-connected neurons. The structure of this model is formulated from extensive prior knowledge of gene ontology. The architecture of GOAE is shown in Fig. 1.

The input layer is genes involved in the scRNA-seq datasets. In hidden layer 1, BP neurons and MF neurons are added based on the biological process and molecular function terms obtained from GO. As shown in Fig. 1, BP and MF neurons are partially connected. Only genes annotated by the corresponding GO term are fed to the GO term neuron.

GOAE consists of two components: encoder and decoder. From the input layer to hidden layer 2 are the

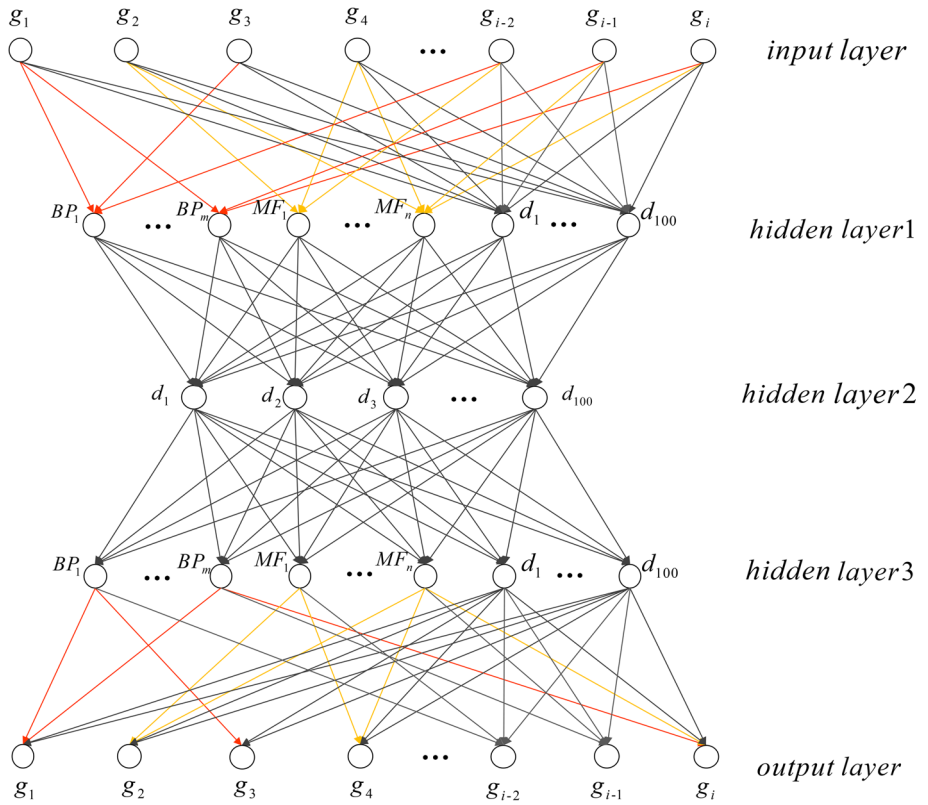

Fig. 1 Gene Ontology Autoencoder (GOAE) model architecture. BP_m and MF_n are the GO-term neurons corresponding to biological process and molecular function category respectively. These neurons partially connected with the input layer. Node d is the full connected neurons. g_i represents the input gene in the given dataset

encoder. The decoder part is exactly a mirror of the encoder part, which from hidden layer 2 to the output layer. Let x_i be the output of the ith hidden layer. The forward propagation of the neural network is:

$$x_i = f(W_i x_{i-1} + b_i), \tag{3}$$

where W_i represents the weight matrix of the edge from $i-1$ th layer to ith layer in the neural network, b_i is the bias of each ith hidden layer node, $f(\cdot)$ is the activation function. We choose tanh function in our case, which is:

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}. \tag{4}$$

In this GOAE model, we use the mean square error as a loss function. Let x_{0j} be the input vector of sample j, and x_{4j} is the output vector. n represents the number of training sample. The loss function is defined as follows:

$$loss = \frac{1}{n}\sum \|x_{0j} - x_{4j}\|^2. \tag{5}$$

After several training epochs, the hidden layer 2 could be a low-dimension space of the input data.

Since the encoder and decoder are completely symmetric, both input layer and output layer are partial connection.

After training GOAE model, the hidden layer 2 could be used as the low-dimension representation of a cell. Then we can use a clustering method, (in our case, kmeans++), for the clustering of single cells.

Architecture of supervised model (GONN)

A supervised dimensionality reduction model may also be needed in single cell clustering or retrieval [15]. Similar to the GOAE model, we replace the hidden layer1 neurons of the neural network with GO term nodes, which are partial-connected to the input layer neurons that represents the genes. In the GONN model, another hidden layer with 100 fully-connected neurons are added (see Fig. 2). After the training phase, the hidden layer with 100 fully-connected neurons is considered as the low dimensional representation of the input.

At the output layer, softmax function is used for classification. Softmax function is defined as:

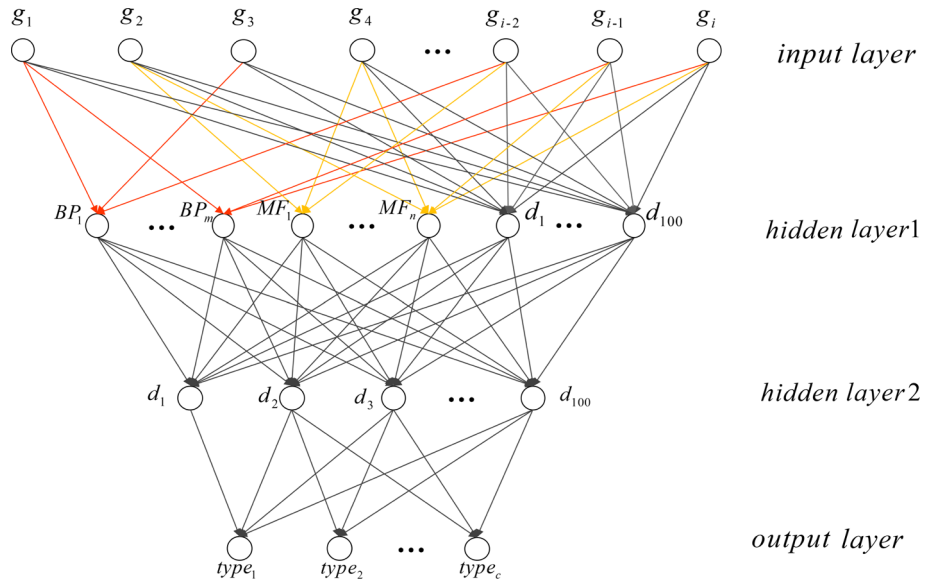

Fig. 2 GONN model architectures. $type_c$ represents different cell types as the label in the output layer. BP_m, MF_n, d and g_i are illustrated in Fig. 1

$$softmax(x) = \left[\frac{\exp(x_1)}{\sum_{i=1}^{c}\exp(x_i)} \cdots \frac{\exp(x_c)}{\sum_{i=1}^{c}\exp(x_i)}\right]^T, \tag{6}$$

where x is the input vector of output layer and c is the number of all cell types. Based on softmax activation function, we can obtain the probability vector that a cell is classified into different cell types. Finally, we use top-1 method (the label which has the largest probability) to decide the cell type of a cell. In GONN, the loss is defined as:

$$loss = -\frac{1}{n}\sum_j \left[y_j \ln y'_j + (1-y_j)\ln(1-y'_j)\right] + \frac{\lambda}{2n}\sum_w w^2, \tag{7}$$

where n is the number of samples in the training dataset. The first part of Eq. 7 is cross entropy. y_j and y'_j represent the desired output and the predicted output of sample j respectively. The second part is L2 regularization, where λ is the L2 regularization coefficient. w represents the training parameter vector. We combine cross entropy and L2 regularization to avoid overfitting and optimize parameters.

After training GONN models by known label cells, we extract the information of the last hidden layer(hidden layer2) as the low-dimension representation. Then we can use a clustering method, (in our case, kmeans++), for the clustering of single cells.

Evaluation criteria

We use the adjusted rand index(ARI) [29] to compare the clustering results of single cells with the true labels. ARI score can measure the similarity between two clustering results. It is defined as follows. Let $X = \{X_1, \ldots, X_r\}$ and $Y = \{Y_1, \ldots, Y_s\}$ be two different clustering results. n_{ij} represents the number of objects in common between X_i and Y_j. Let $a_i = \sum_j n_{ij}$ and $b_j = \sum_i n_{ij}$, the ARI is defined as follow:

$$\begin{aligned} ARI &= \frac{Index - ExpectedRandIndex}{MaxRandIndex - ExpectedRandIndex} \\ &= \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]/\binom{n}{2}}. \end{aligned} \tag{8}$$

The scale of ARI score is between -1 and 1. The higher the ARI score is, the more similar two clustering results are.

Furthermore, normalized mutual information(NMI) [30] is also used for evaluation. NMI uses the concept of information entropy to compare different clustering results. NMI score is calculated as follows:

$$NMI = \frac{I(X,Y)}{\sqrt{H(X)H(Y)}}. \tag{9}$$

$H(X)$ is the entropy of X, which is calculated as follows:

$$H(X) = -\sum_i \frac{a_i}{N}\log\frac{a_i}{N}. \tag{10}$$

$I(X,Y)$ is the mutual information between X and Y, which is calculated as follows:

$$I(X,Y) = \sum_i\sum_j \frac{n_{ij}}{N}\log\frac{n_{ij}/N}{a_i b_j/N^2}, \tag{11}$$

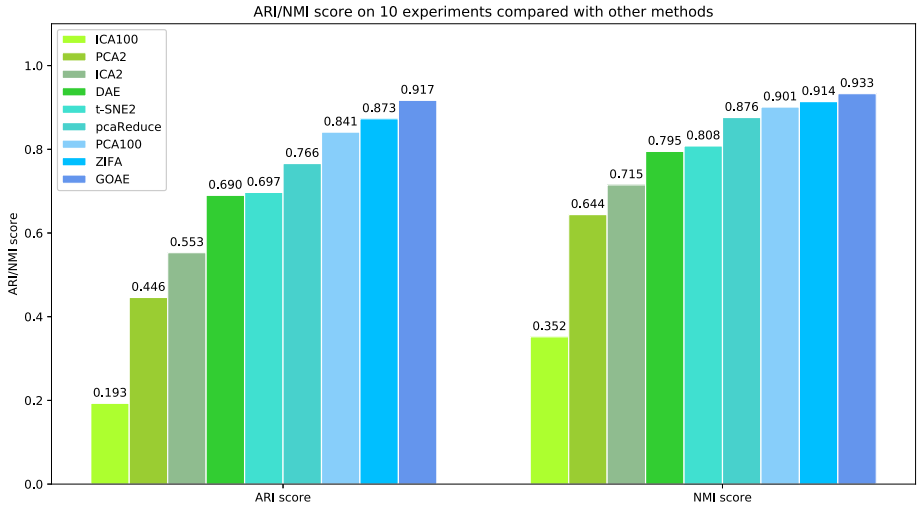

Fig. 3 Average ARI and MNI scores on 10 experiments of different unsupervised methods on clustering 11 types of human cells. The number after methods means n components. i.e.PCA2 means using PCA method which we set 2 components

where $N = \sum_i \sum_j n_{ij}$. NMI scores are between 0 and 1. The higher the NMI score is, the more similar two clustering results are. In the following evaluations, we run each experiment 10 times and calculate their average scores as final results.

Data preparation

We evaluate our models on three scRNA-seq datasets. The first dataset is a human scRNA-seq data from [31]. In our experiment, 300 cells involving 11 cell types are used. The involved cell types are listed as follows: CRL-2338(epithelial), CRL-2339(lymphoblastoid), BJ(fibroblast from human foreskin), GW(gestational 16, 21, 21+3 weeks from fetal cortex), HL60(myeloid from acute leukemia), iPS(pluripotent), K562(myeloid from chronic leukemia), Kera(foreskin keratinocyte) and NPC(neural progenitor cells). We remove the genes that have missing values in these cell types. Eigth thousand six hundred eighty six genes are involved in the evaluation dataset. The second dataset is obtained from [15]. It integrates three mus musculus scRNA-seq datasets [14, 32, 33], which contains 402 cells involving 16 cell types. Similarly, after removing the genes with missing values, 9437 genes are included in the evaluation dataset. The third dataset is also a mus musculus dataset from [15], which has more than 17,000 single-cell RNA-seq data from different 31 datasets. We use this dataset to evaluate cell type assignment. The gene ontology data is downloaded from http://www.geneontology.org/.

Results and discussion

We test our models on two different scRNA-seq datasets. We compare our methods with two supervised methods (i.e. NN(ppi/tf) [15] and NN(dense)) and six unsupervised methods(i.e. PCA [12], t-SNE [17], ICA [34], pcaReduce [35], ZIFA [18], DAE [36]). We set batch size as 64, epoch number as 100, learning rates as 1e-3 for GOAE model. We set the batch size as 64, epoch number as 200, learning rates as 0.2 for GONN model. For NN(dense) model, it has the same architecture as the two-layer GONN model but without partial connection between the input layer and hidden layer1. The NN(dense) model is used to test whether combining GO information can improve the supervised model. The DAE model is used to test whether the addition of GO information can improve the unsupervised neural network model. We also compare our model with other unsupervised methods. In all tests, we use *kmeans* + + for clustering based on different low-dimensional representations from different dimensionality reduction methods. The models are implemented using Python 3.6 and tensorflow 1.4.1 package.

Performance evaluation on human scRNA-seq dataset

We test GOAE model (Fig. 1) and GONN model (Fig. 2) for clustering of human cells. 1174 GO terms satisfy the criteria described in 2.1 subsection. These terms are used in the GOAE and GONN model.

Table 2 Average ARI scores of 10 experiments compared with other supervised model on human scRNA-seq dataset

Number of clusters	2	4	6
NN(dense)[1]	0.9123	0.7806	0.7427
NN(ppi/tf)[1]	0.9925	0.8696	0.7542
GONN[1]	**0.9975**	**0.9036**	**0.8189**

For NN(dense) model, we set epoch number=200, learning rate=0.2. For NN(ppi/tf) model, the parameters are same as [15] epoch number=100, learning rate=0.1. For GONN model, we set epoch number=200, laerning rate=0.2

Table 3 Average NMI scores of 10 experiments compared with other supervised model on human scRNA-seq dataset

Number of clusters	2	4	6
NN(dense)	0.9008	0.8367	0.8434
NN(ppi/tf)	0.9873	0.9056	0.8243
GONN	**0.9918**	**0.9179**	**0.8803**

See Table 2 for the hyper parameter selection of each model. Numbers in bold indicate the best performance

In the unsupervised test, all the unsupervised models are applied to the whole data set. All 11 types of cells are involved. Overall, GOAE performs the best among all tested methods. Similar with the experiment design in [15], several possible parameters (number of components) are tested for PCA and ICA method. We reduce the dimension of all data and using kmeans++ method to cluster all 11 cell types data. Figure 3 shows that GOAE perfects the best among all tested methods. The ARI and NMI score of GOAE are 0.917 and 0.933 respectively, while the scores of the runner-up method ZIFA are 0.873 and 0.914 respectively. The experiment result indicates that combining Gene Ontology and autoencoder can improve the performance of clustering of single cells.

For the supervised model, we compare GONN with the state-of-art method NN(ppi/tf) [15] and the original neural network model (NN). We apply the same experimental protocol used in [15]. The cell types not used in the training phase are used as the test set. There are 11 cell types involved in this data set. We randomly select 2, 4 and 6 cell types as the test set in the evaluation test.

Overall, GONN method performs better than other methods (Tables 2 and 3). With the increase of the number of cell types in the test set, the clustering task becomes more challenging. The result shows that GONN performs the best when the number of cell types equals to 2, 4 and 6. Furthermore, when the number of cell types is 6, the ARI score of GONN is 0.8189, which is significantly higher than the runner-up method (Table 2). Unsurprisingly, GONN method also achieves the highest NMI score. The NMI score of GONN is 0.8803 even when the number of cell types is 6, while the value of the second best method is 0.8434.

Figure 4 is the 2D visualization of low dimensional representation based on GONN and GOAE. We use t-SNE as the visualization tool. It is shown that the single cells are partitioned into different clusters based on GONN and GOAE, indicating that GONN and GOAE can learn a low dimensional representation for single cell data.

Performance evaluation on mus musculus dataset

Similar with evaluation test on the human dataset, we also test these models on mus musculus dataset that contains 16 cell types. For unsupervised models, we randomly select 2, 4, 6, 8, 10 and 12 cell types as test sets. For supervised models, since sufficient training set is necessary, we only randomly select 2, 4, 6 and 8 cell types as test sets. The rest of data are used as the training set. For GOAE and GONN model, 854 GO terms satisfy the criteria described in 2.1 subsection.

As shown in Tables 4 and 5, for the unsupervised model, GOAE achieves the highest performance on datasets with different numbers of cell types. The average of ARI scores of GOAE on all datasets is 0.7671, which is around 0.03 higher than the runner-up method DAE. More details are shown in Table 4. The trend of NMI scores is similar to ARI scores. GOAE can achieve the highest NMI scores on datasets with different numbers of cell types.

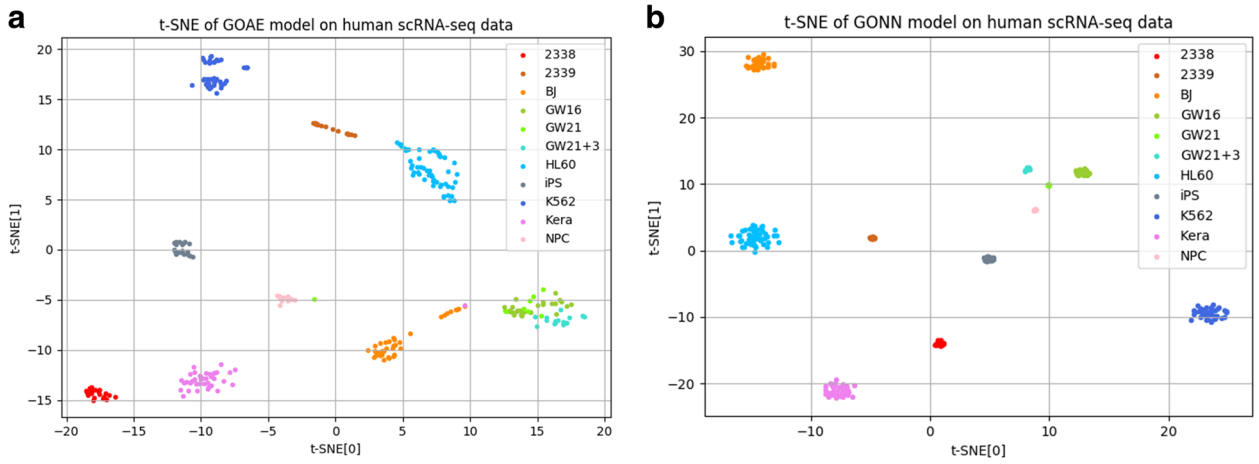

Fig. 4 2D t-sne visualizations for our model on human scRNA-seq data. **a** is the dimension reduction result on GOAE model. **b** is the dimension reduction result on GONN model

Table 4 Average ARI score of 10 experiments compared with other methods on mus musculus dataset

Number of clusters	2	4	6	8	avg1	10	12	avg2
NN(dense)	0.9562	0.8231	0.6909	0.6832	0.7884	/	/	/
NN(ppi/tf)	0.9288	0.7983	0.7077	0.6630	0.7745	/	/	/
GONN	**0.977**	**0.9199**	**0.7934**	**0.7599**	**0.8626**	/	/	/
PCA2	0.9583	0.606	0.49	0.4489	0.6258	0.4184	0.3819	0.5505
ICA2	0.8296	0.5798	0.4786	0.4656	0.5584	0.4293	0.4026	0.5209
t-SNE2	0.4072	0.5223	0.5413	0.596	0.5167	0.5758	0.5725	0.5359
PCA10	0.9583	0.6373	0.5926	0.6073	0.6989	0.5761	0.5604	0.6553
ICA10	0.0535	0.4445	0.4119	0.5502	0.365	0.5231	0.5053	0.4148
PCA100	0.8707	0.7749	0.5792	0.5634	0.6971	0.5428	0.6031	0.6557
ICA100	0.281	0.075	0.0098	0.0307	0.0834	0.0324	0.0694	0.0726
ZIFA	-0.0143	0.2115	0.4275	0.5847	0.3024	0.6151	**0.6212**	0.4076
pcaReduce	0.6476	0.5604	0.5358	0.4777	0.5553	0.4399	0.3888	0.5084
DAE	**0.9758**	0.8435	0.718	0.698	0.8088	0.6226	0.584	0.7403
GOAE	0.967	**0.8614**	**0.7875**	**0.7381**	**0.8385**	**0.6401**	0.6085	**0.7671**

The number after other unsupervised methods means n components. i.e.PCA2 means using PCA method which we set 2 components. For DAE model, we set epoch number as 200 and learning rate as 1e-3. For GOAE model, we set epoch number as 100 and learning rate as 1e-3. The parameters in other NN-based models are shown in Table 2. Avg1 is the average ARI score of the formal four cluster results, while avg2 ARI score is the average of all 2,4,6,8,10 and 12 cluster results. The highest values are shown in boldface

The complexity of the problem increases with the increase in the number of cell types. When the number of cell types is 8, the NMI score of GOAE is 0.8545 that is 0.04 higher than the runner-up method DAE. The evaluation test on mus musculus dataset indicates that combining gene ontology with neural network can improve the performance of single cell RNA-seq data clustering.

For the supervised model, GONN performs better than other compared methods. The ARI score decreases with the increase in the number of cell types involved in the test set. GONN can achieve a high ARI score (0.7599) even the number of cell types is 8, while the value of runner-up method is 0.6832. Similarly, GONN also achieves the highest NMI score in all tested methods. The average NMI score of different datasets is 0.9103, which is significantly higher than $NN(dense)$ and $NN(ppi/tf)$ method. The corresponding values of $NN(dense)$ and $NN(ppi/tf)$ are 0.8623 and 0.8496 respectively.

Table 5 Average NMI scores on 10 experiments compared with other methods on mus musculus dataset

Number of clusters	2	4	6	8	avg1	10	12	avg2
NN(dense)	0.9348	0.8794	0.8179	0.8171	0.8623	/	/	/
NN(ppi/tf)	0.9083	0.8673	0.8119	0.811	0.8496	/	/	/
GONN	**0.9688**	**0.9366**	**0.8721**	**0.8635**	**0.9103**	/	/	/
PCA2	0.9527	0.727	0.673	0.6756	0.7571	0.6567	0.6408	0.7210
ICA2	0.8374	0.6966	0.6635	0.6828	0.7201	0.6632	0.6553	0.7261
t-SNE2	0.4025	0.6101	0.6778	0.734	0.6061	0.7432	0.7531	0.6467
PCA10	0.9527	0.7574	0.7349	0.758	0.8008	0.7425	0.7382	0.7835
ICA10	0.1367	0.6224	0.6095	0.7196	0.5221	0.709	0.6965	0.5737
PCA100	0.8656	0.8509	0.7675	0.7812	0.8163	0.7794	0.8215	0.8118
ICA100	0.2186	0.1319	0.1173	0.142	0.15245	0.1539	0.249	0.1665
ZIFA	0.0548	0.3721	0.6267	0.7716	0.4563	0.8188	**0.8271**	0.5611
pcaReduce	0.6645	0.7115	0.6853	0.6649	0.6816	0.6499	0.6247	0.6689
DAE	**0.9621**	0.8877	0.8172	0.8194	0.8716	0.8047	0.7957	0.8478
GOAE	0.9544	**0.9076**	**0.8693**	**0.8545**	**0.8965**	**0.8206**	0.8018	**0.8680**

The highest values are shown in blodface. See Table 4 for the more details

Table 6 Average ARI scores of 10 experiments for GONN model when select different U_{ij} score

U_{ij} score	Number of clusters			
	2	4	6	8
0.3	0.936	0.8578	0.752	0.7296
0.4	**0.9952**	0.874	0.7492	0.7209
0.5	0.977	**0.8866**	**0.7579**	**0.7356**
0.6	0.9818	0.8701	0.7312	0.7049
0.7	0.9670	0.8489	0.7317	0.6824

Numbers in bold indicate the best performance

Table 7 Average ARI scores of 10 experiments for GONN model when select different H_i score

H_i score	Number of clusters			
	2	4	6	8
0.05	0.9653	0.8444	0.7571	0.6846
0.075	0.967	0.8644	0.755	0.7123
0.1	0.9852	**0.8818**	**0.7662**	**0.7164**
0.125	0.967	0.8489	0.7585	0.7006
0.15	**0.9952**	0.8245	0.7554	0.7109

Numbers in bold indicate the best performance

Effect of GO terms

One of the major contributions of our work is to add GO terms as neurons in the neural networks. To test whether the GO terms are selected appropriately, we re-run GONN and GOAE by varying the GO terms involved in the model. We use the mus musculus dataset on this test.

To determine the threshold selection for the U_{ij} and H_i scores, we varied one parameter and fix other parameters to conduct experiments on GONN (see Tables 6 and 7). The evaluation test shows that GONN can achieve the highest performance when the unique score and high expression score are set as 0.5 and 0.1 respectively.

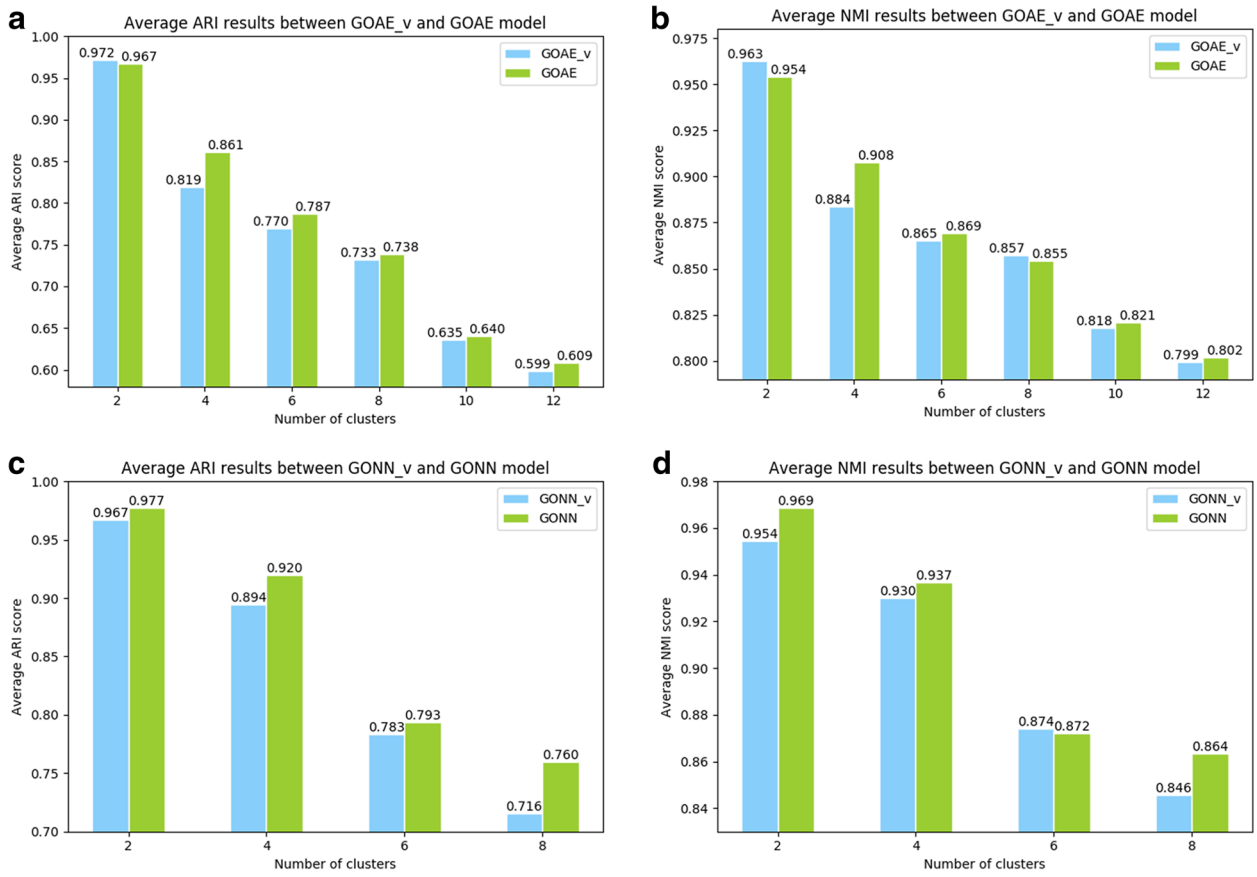

Fig. 5 Performance evaluation by selecting different numbers of GO terms for GOAE and GONN model. **a**, **b** Average ARI and NMI results between $GOAE_v$ model and *GOAE* model. $GOAE_v$ represent GOAE model without selection of GO terms. **c**, **d** Average ARI and NMI results between $GONN_v$ model and *GONN* model. $GONN_v$ represent GONN model without selection of GO terms. GOAE models in (**a**) and (**b**) select different hyperparameters. Before GO terms selection, it selects epoch number=300, learning rate=1e-4. After selection, it selects epoch number=100, learning rate=1e-3. All GONN models in (**c**) and (**d**) all select epoch number=200 and learning rate=0.2. All these hyperparameters selection are achieve their best across a lots of hyperparameter groups

As described in subsection 2.1, we remove the redundancy GO terms and GO terms with low expression scores. In this test, we create $GONN_v$ and $GOAE_v$ where the redundancy and low-diversity GO terms are not removed. In $GONN_v$ and $GOAE_v$, 1486 GO terms are involved, while only 854 GO terms involved in GONN and GOAE. Figure 5a and b show that GONN is clearly better than $GONN_v$, indicating that selecting appropriate GO terms contributes to the performance and this step has been appropriately designed. Similarly, Fig. 5c and (d) show that GOAE is clearly better than $GOAE_v$. Particularly, on the datasets with 8 and 10 cell types, the average ARI of GOAE are about 2-3% higher than $GOAE_v$.

Table 8 Highly ranked GO-term nodes for some cell types used for traing GOAE and GONN models

Model	Cell type	GO term	GO function
GOAE	ES	GO:0043008	ATP-dependent protein binding [37]
	ES	GO:0032409	Regulation of transporter activity [37]
GONN	ES	GO:0022417	Protein maturation by protein folding [37]
	ES	GO:0140101	Catalytic activity, acting on a tRNA [37]
	BMDC	GO:0099590	Neurotransmitter receptor internalization [40]
	BMDC	GO:0050881	Musculoskeletal movement [40]
	Zygote	GO:0001892	Embryonic placenta development [38]
	Early 2cell	GO:0032552	Deoxyribonucleotide binding [41]

Functional analysis on hidden layer nodes

For GOAE model, we train the model using samples of a certain cell type. Then, we could also obtain the top 10 highest GO-term nodes of the hidden layer. We select 8cell, 16cell, ES, earlyblast, and lateblast in this test, since training the GOAE model requires a sufficient amount of samples. For GONN model, we multiply the weight matrices $W2$ and $W3$ to represent the degree of importance between each cell type and the GO terms in the hidden layer 1. For each cell type, we selected the top-10 important GO terms for analysis. Table 8 shows some of the highly weighted GO-term nodes in the GOAE and GONN models. For example, regulation of transporter activity (GO:0032409) is mainly associated with ES(embryonic stem cell) [37], and embryonic placenta development (GO:0001892) is always relative with zygote cell [38].

Cell type assignment

Another important application in single cell analysis is cell type assignment. To verify the effectiveness of our model in cell assignment and retrieval. We use a mus musculus dataset from Lin et al. paper [15], which has more than 17,000 single cells from different 31 datasets. We designed experiment according to [15].

To measure the results of cell type assignment, we calculate the percentage of the correctly predicted cell types by using top K nearest neighbors(K=100). Nine cell types are involved in the experiment, including 2 cell, 4 cell, 8 cell, zygote, embryonic stem cell(ESC), neurons, thymus, spleen and hematopoietic stem cell(HSC). Mean of average precision(MAP) [15, 39] is used to measure the assignment performance.

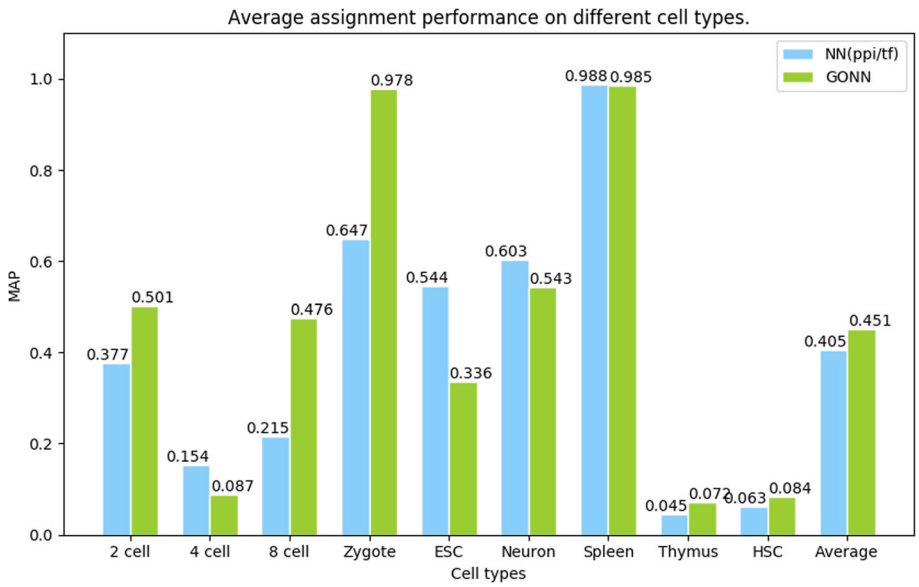

Fig. 6 Average assignment performance on different cell types. The scores are mean of average precision

We compare our GONN model with NN(ppi/tf) model, the results are shown in Fig. 6. Our model GONN performs better in 2 cell, 8 cell, zygote cell types. Besides, GONN has higher average of MAP than NN (ppi/tf).

Conclusions

In this paper, we combine neural networks with Gene Ontology for reducing the dimensions of scRNA-seq data, which can improve the clustering of scRNA-seq data. We propose two models GOAE and GONN that are unsupervised and supervised model respectively.

The proposed model mainly contains two key components: the selection of significant GO terms and combination GO terms with the neural network-based model. When selecting important GO terms, it is crucial to choose the appropriate thresholds. If the threshold is not properly selected, deleting too much or too few GO terms will affect the final result.

Performance evaluation on two datasets shows that GONN and GOAE perform better than existing state-of-art dimensionality reduction methods for scRNA-seq data.

Abbreviations

ARI: Adjusted rand index; DAE: Denoising autoencoder; GO: Gene ontology; GOAE: Gene ontology autoencoder; GONN: Gene ontology neural network; MAP: Mean of average precision; NMI: Normalized mutual information; PCA: Principle component analysis; PPI: Protein protein interaction; SC3: Single-cell consensus clustering; scRNA-seq: Single cell RNA sequence; SNN-Cliq: Shared nearest neighbor Cliq; T-SNE: T-distributed stochastic neighbor embedding; ZIFA: Zero inflated factors analysis

Acknowledgements

We thank all the anonymous reviewers.

Funding

The publication costs for this article were funded by the corresponding author's institution. This work was supported by National Natural Science Foundation of China (No. 61702421, 61332014, 61772426), China Postdoctoral Science Foundation (No. 2017M610651), China Postdoctoral Science Foundation (No. 2017BSHTDZZ11), Fundamental Research Funds for the Central Universities (No. 3102018zy033).

About this supplement

This article has been published as part of *BMC Bioinformatics Volume 20 Supplement 8, 2019: Decipher computational analytics in digital health and precision medicine*. The full contents of the supplement are available online at https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-20-supplement-8.

Authors' contributions

JP and XS designed the algorithm; XW implemented the algorithm; JP and XW wrote this manuscript. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Author details

[1]School of Computer Science, Northwestern Polytechnical University, 710072 Xi'an, China. [2]Key Laboratory of Big Data Storage and Management, Northwestern Polytechnical University, Ministry of Industry and Information Technology, 710072 Xi'an, China. [3]Centre for Multidisciplinary Convergence Computing, School of Computer Science, Northwestern Polytechnical University, 710072 Xi'an, China.

Published: 10 June 2019

References

1. Wang Z, Gerstein M, Snyder M. Rna-seq: a revolutionary tool for transcriptomics. Nat Rev Genet. 2009;10(1):57.
2. Cheng L, Hu Y, Sun J, Zhou M, Jiang Q. Dincrna: a comprehensive web-based bioinformatics toolkit for exploring disease associations and ncrna function. Bioinformatics. 2018;34(11):1953–56.
3. Cheng L, Wang P, Tian R, Wang S, Guo Q, Luo M, Zhou W, Liu G, Jiang H, Jiang Q. Lncrna2target v2.0: a comprehensive database for target genes of lncrnas in human and mouse. Nucleic Acids Res. 2019;47(D1):D140-D144.
4. Stegle O, Teichmann SA, Marioni JC. Computational and analytical challenges in single-cell transcriptomics. Nat Rev Genet. 2015;16(3):133.
5. Raj A, van Oudenaarden A. Nature, nurture, or chance: stochastic gene expression and its consequences. Cell. 2008;135(2):216–26.
6. Kolodziejczyk A, Kim JK, Svensson V, Marioni J, Teichmann S. The technology and biology of single-cell rna sequencing. Mol Cell. 2015;58(4):610–20.
7. Hu Y, Tianyi Z, Tianyi Z, Ying Z, Liang C. Identification of alzheimer's disease-related genes based on data integration method. Front Genet. 2018;9:703.
8. Tang F, Barbacioru C, Wang Y, Nordman E, Lee C, Xu N, Wang X, Bodeau J, Tuch BB, Siddiqui A. mrna-seq whole-transcriptome analysis of a single cell. Nat Methods. 2009;6(5):377–82.
9. Wu AR, Neff NF, Kalisky T, Dalerba P, Treutlein B, Rothenberg ME, Mburu FM, Mantalas GL, Sim S, Clarke MF. Quantitative assessment of single-cell rna-sequencing methods. Nat Methods. 2014;11(1):41–46.
10. Jaitin DA, Kenigsberg E, Keren-Shaul H, Elefant N, Paul F, Zaretsky I, Mildner A, Cohen N, Jung S, Tanay A. Massively parallel single-cell rna-seq for marker-free decomposition of tissues into cell types. Science. 2011;343(6172):776–9.
11. Chung W, Eum HH, Lee HO, Lee KM, Lee HB, Kim KT, Ryu HS, Kim S, Lee JE, Park YH. Single-cell rna-seq enables comprehensive tumour and immune cell profiling in primary breast cancer. Nat Commun. 2017;8:15081.
12. Wold S, Esbensen K, Geladi P. Principal component analysis. Chemometr Intell Lab Syst. 1987;2(1):37–52.
13. Buettner F, Natarajan KN, Casale FP, Proserpio V, Scialdone A, Theis FJ, Teichmann SA, Marioni JC, Stegle O. Computational analysis of cell-to-cell heterogeneity in single-cell rna-sequencing data reveals hidden subpopulations of cells. Nat Biotechnol. 2015;33(2):155–60.
14. Shalek AK, Satija R, Adiconis X, Gertner RS, Gaublomme JT, Raychowdhury R, Schwartz S, Yosef N, Malboeuf C, Lu D. Single-cell transcriptomics reveals bimodality in expression and splicing in immune cells. Nature. 2013;498(7453):236.
15. Lin C, Jain S, Kim H, Barjoseph Z. Using neural networks for reducing the dimensions of single-cell rna-seq data. Nucleic Acids Res. 2017;45(17):156.
16. Li X, Chen W, Chen Y, Zhang X, Gu J, Zhang MQ. Network embedding-based representation learning for single cell rna-seq data. Nucleic Acids Res. 2017;45(19):166.
17. Maaten L, Hinton G. Visualizing data using t-sne. J Mach Learn Res. 2008;9(2605):2579–605.
18. Yau C, Pierson E. Dimensionality reduction for zero-inflated single cell gene expression analysis. Genome Biol. 2015;16(1):241.
19. Xu C, Su Z. Identification of cell types from single-cell transcriptomes using a novel clustering method. Bioinformatics. 2015;31(12):1974–80.
20. Kiselev VY, Kirschner K, Schaub MT, Andrews T, Yiu A, Chandra T, Natarajan KN, Reik W, Barahona M, Green AR, et al. Sc3: consensus clustering of single-cell rna-seq data. Nat Methods. 2017;14(5):483.
21. Ma J, Yu MK, Fong S, Ono K, Sage E, Demchak B, Sharan R, Ideker T. Using deep learning to model the hierarchical structure and function of a cell. Nat Methods. 2018;15(4):290.
22. Carbon S, Ireland. A, Mungall CJ, Shu SQ, Marshall B, Lewis S, Hub TA. Amigo: online access to ontology and annotation data. Bioinformatics. 2009;25(2):288–9.

23. Peng J, Hui W, Shang X. Measuring phenotype-phenotype similarity through the interactome. BMC Bioinformatics. 2018;19(5):114.
24. Peng J, Xue H, Shao Y, Shang X, Wang Y, Chen J. A novel method to measure the semantic similarity of hpo terms. Int J Data Min Bioinforma. 2017;17(2):173–88.
25. Melott JM, Weinstein JN, Broom BM. Pathwaysweb: a gene pathways api with directional interactions, expanded gene ontology, and versioning. Bioinformatics. 2016;32(2):312–4.
26. Peng J, Zhang X, Hui W, Lu J, Li Q, Liu S, Shang X. Improving the measurement of semantic similarity by combining gene ontology and co-functional network: a random walk based approach. BMC Syst Biol. 2018;12(2):18.
27. Pesaranghader A, Matwin S, Sokolova M, Beiko RG. simdef: definition-based semantic similarity measure of gene ontology terms for functional similarity analysis of genes. Bioinformatics. 2016;32(9):1380–7.
28. Peng J, Wang T, Wang J, Wang Y, Chen J. Extending gene ontology with gene association networks. Bioinformatics. 2015;32(8):1185–94.
29. Hubert L, Arabie P. Comparing partitions. J Classif. 1985;2(1):193–218.
30. Vinh NX, Epps J, Bailey J. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. Cambridge: JMLR.org; 2010, pp. 1073–80.
31. Pollen AA, Nowakowski TJ, Shuga J, Wang X, Leyrat AA, Lui JH, Li N, Szpankowski L, Fowler B, Chen P. Low-coverage single-cell mrna sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. Nat Biotechnol. 2014;32(10): 1053–8.
32. Sasagawa Y, Nikaido I, Hayashi T, Danno H, Uno KD, Imai T, Ueda HR. Quartz-seq: a highly reproducible and sensitive single-cell rna sequencing method, reveals non-genetic gene-expression heterogeneity. Genome Biol. 2013;14(4):3097.
33. Deng Q, Ramsköld D, Reinius B, Sandberg R. Single-cell rna-seq reveals dynamic, random monoallelic gene expression in mammalian cells. Science. 2014;343(6167):193–6.
34. Comon P. Independent Component Analysis, a New Concept? Oxford: Elsevier North-Holland, Inc.; 1994, pp. 287–314.
35. Žurauskienė J, Yau C. pcareduce: hierarchical clustering of single cell transcriptional profiles. BMC Bioinformatics. 2016;17(1):140.
36. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res. 2010;11(12): 3371–408.
37. Sene KH, Porter CJ, Palidwor G, Pereziratxeta C, Muro EM, Campbell PA, Rudnicki MA, Andradenavarro MA. Gene function in early mouse embryonic stem cell differentiation. BMC Genomics. 2007;8(1):85.
38. Pawel K, Vijay C, Carsten P. Simulating the mammalian blastocyst - molecular and mechanical interactions pattern the embryo. PloS Comput Biol. 2011;7(5):1001128.
39. Zhang E, Yi Z. Average Precision. Boston: Springer; 2009. p. 192–93.
40. Cruz DSGD, Lima APND, Neto JP, Massoco C. Effects of unilateral cervical vagotomy on murine dendritic cells. Am J Immunol. 2015;11(2):48–55.
41. Ko MSH, Zalzman M, Sharova LV. Methods for enhancing genome stability and telomere elongation in embryonic stem cells. US; 2015. U.S. Patent Application 14/259,600, filed August 21, 2014.