Deep-Cloud: A Deep Neural Network-Based Approach for RNA-seq Gene Expression Analysis

Qinyu Ge (geqinyu@seu.edu.cn)
Southeast University

Ying Zhou
Southeast University

Erteng Jia
Southeast University

YHuajuan Shi
Southeast University

Zhiyu Liu
Southeast University

Yuqi Sheng
Southeast University

Min Pan
Southeast University

Jing Tu
Southeast University

Zuhong Lu
Southeast University

Research Article

Keywords: Deep-cloud, RNA-seq, differential expressed genes, deep neural networks

Posted Date: June 14th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1663687/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Abstract

Background

Presently, the field of differential expression analysis of RNA-seq data is still in its infancy with new approaches constantly being proposed. Combining deep neural networks to explore gene expression information in RNA-seq data provides a novel possibility in this field.

Results

This study focuses on the development of a deep neural network-based RNA-seq gene expression analysis model, named Deep-cloud. Its main advantage is not only the excellent construction of deep learning network models using convolutional neural network and long short-term memory to predict the gene expression of RNA-seq, but also the more in-depth analysis of the differential gene expression between disease and normal groups of transcriptome data by combining the statistical methods of cloud model.

Conclusion

Comparing the results of this data analysis with those obtained by traditional differential gene analysis software such as DESeq2 and edgeR, the Deep-cloud model further improves the sensitivity of obtaining differential expressed genes. Overall, the proposed Deep-cloud paves a new pathway for the biomedical field.

Background

RNA-seq data are increasingly available with the continuous development of high-throughput sequencing technology. The interpretation of RNA-seq data is the basis and starting point for gene function and structure studies, and is a necessary step to reveal the molecular composition of cells and tissues. [1] With the obvious advantages of RNA-seq over traditional sequencing methods, it is widely used for differential gene expression analysis, new transcript discovery, variable shear and others. Its application is becoming more widespread as the technology advances and the cost decreases. RNA-seq is currently the first choice of technology for performing differential gene expression experiments, and one of its most common analytical tasks is to identify genes that are deferentially expressed between experimental conditions. However, it is not clear which statistical tools are best suited to analyze the data. [2] Moreover, the characterization of RNA-seq data has not been fully established, and further studies are needed to understand how these data respond to differential expression analysis. [3] There is an increasing demand for the analysis of single-cell transcriptome sequencing results, especially in the contemporary era of the rise of single-cell spatial transcriptomics. [4] Specifically, a number of software packages have been developed for differential expression analysis of RNA-seq data, such as DESeq2, edgeR, NBPSeg, TSPM,
baySeq, EBSeq, NOISeq, SAMseq and so on. [5–10] However, each of these methods has advantages and disadvantages. For example, DESeq2 is usually too conservative and edgeR tends to be too liberal at fairly large sample sizes. In addition, Cuffdiff 2 allows for robust differential expression analysis at both gene and isoform level resolution. Transcript resolution measurements using RNA-seq can greatly simplify the problem by eliminating unexpressed transcripts and isolating enriched transcripts. [11] Alternatively, there are methods that combine multiple techniques for analysis. For example, DESeq2 and edgeR-based R package SARTools can be used to perform comprehensive differential analysis of RNA-Seq data. [12] Overall, the field of differential expression analysis of RNA-seq data is still in its infancy, and new methods are being proposed all the time. To date, there is no general consensus on which method performs best in a given situation. Moreover, the results obtained from RNA-seq gene expression analysis are of great value for studies on diagnosis, typing, treatment and prognosis of diseases. [13, 14] Therefore, continued development of new tools for RNA-seq gene expression analysis is absolutely necessary to obtain valuable information. Currently, harvesting gene expression information from RNA-seq data using deep neural networks offers a novel possibility for the field, promising to discover information that previous approaches have failed to find.

With the rapid development of life sciences and computer science, artificial intelligence technologies represented by machine learning have been gradually developed along with the exponential growth of biological data. The growth of biological data has greatly enriched the data resources of bio-informatics in terms of quality and quantity, providing a data base for solving the mystery of life. The artificial intelligence technology becomes a powerful tool for mining these data. Deep learning, a new field of research, is gaining more and more attention. [15] Deep learning is increasingly being used in major fields including bio-medicine due to its powerful capabilities. Deep neural network models have been used to solve the problem of low accuracy and poor precision of traditional algorithms in screening differentially expressed genes and function prediction. [16] There are also studies that use deep learning for mining gene expression results like plants and cotton. [17, 18] Deep learning prediction frameworks have been shown to be excellent for genomics mutation detection and for inferring causal mutations in diseases and for predicting disease risk. [19] The machine learning models have also shown good performance in biomarker prediction for clinical diseases such as cancer. [20, 21] These indicate that deep learning network models have great potential when used to explore useful information of gene expression in RNA-seq data. However, there is still a lack of an accurate and efficient method of deep learning to fully exploit the gene expression information in RNA-seq data. There is no consensus on which method is most appropriate or which method can ensure the validity of the results in terms of robustness, accuracy and comprehensibility. This topic in bio-informatics research is still under development.

Our research focuses on the development of a deep neural network-based model for RNA-seq gene expression analysis, named Deep-cloud. (Fig. 1) The main advantage is not only the excellent construction of deep learning network models using convolutional neural network (CNN) and long short-term memory (LSTM) to predict RNA-seq gene expression, but also the deeper analysis of transcriptome data for differential gene expression between disease and normal groups by combining the statistical approach of cloud model. The combination of CNN and LSTM has been shown to have excellent
predictive performance, and its residuals can be used to quantify the variation of differential genes. In addition, it is necessary to combine CNN and LSTM prediction results with cloud model, which is a cognitive model based on probability statistics and fuzzy set theory to achieve bidirectional cognitive transformation between qualitative concepts and quantitative data.[22] The cloud model is used to generate cloud drops from the residuals of CNN and LSTM prediction results, and then the evaluation interval of this data is constructed, which can effectively obtain the differential genes beyond the evaluation interval. Moreover, the results of the CNN and LSTM prediction model and the area under the curve (AUC) of the receiver operating curve (ROC) surrounded by the coordinate axes reached 0.93. Its sensitivity was 89.7% and specificity was 83.4%. The effectiveness of the model was verified by several iterations. Comparing the analysis results of this data with those obtained by traditional differential gene analysis software such as DESeq2 and edgeR, this model further improved the sensitivity of obtaining differential genes. It is able to find more GO or KEGG pathways related to the Parkinson's disease (PD). Overall, Deep-cloud can show great advantages in gene expression analysis of transcriptome data and can be used as another possibility to analyze transcriptome gene expression data.

Results

Prepossessing of RNA-seq data

Instead of using integer counts directly, some software packages represent RNA-seq data by transformed ones. In this study, the relevant number of fragments per kilo-base per million mapped reads (FPKM) was used as input. [23] The goal of this transformation is to normalize the counts according to different library sizes and transcript lengths, as longer transcripts get more reads than shorter transcripts with the same expression level. Instead of much transformation, this model takes the FPKM values as the input to the deep neural network directly. The steps are simpler and more intuitive compared to others such as binary transformation. In addition, mice with Parkinson's disease were used as the experimental group and normal mice with the same experimental and library building conditions were used as the control group in this study. The correlation between each group is high, with p-values > 0.9. (Figure S1)

Neural network model combined with CNN and LSTM

The combination of CNN and LSTM can take full advantage of the respective structural advantages of LSTM and CNN. CNN can enhance the capture of key features while reducing the model complexity due to its unique network structure. The combination of CNN and LSTM can achieve excellent results in gene screening and function prediction. In this study, a combination of CNN and LSTM is used to model and analyze 60,000 sets of gene sequences. The analysis results of the model will be meaningless once the over-fitting phenomenon is caused due to the complex structure of the deep learning network. As a result, the data are divided into a training set and a validation set. The optimal neural network structure is determined by comparing the values of both fitness functions. The global gene sequences are then predicted based on the determined network structure. The corresponding loss function iteration curves of the training set and validation set are shown in Figure 2A. As can be seen from Figure 2A, it tends to be
stable at 100 iterations. The model has strong robustness and generalization ability, which preserves the trained model perfectly.

The ROC curve reflects the combined performance of the model based on sensitivity and specificity. The AUC value is a probability value representing the area under the ROC curve, $AUC \in (0, 1)$. The larger the AUC value, the better the model classifies. The AUC value considers not only the accuracy of the model, but also the sensitivity and specificity. Figure 2B shows the ROC curves and AUC values of the CNN and LSTM model. The accuracy of AUC is low at 0.5-0.7, moderate at 0.7-0.9, and high at 0.9 or above. The larger the area under the curve, the higher the prediction accuracy. The current value of its AUC is 0.93, which indicates that the method has high prediction accuracy. And the model has high sensitivity and specificity with the values of 0.897 and 0.834, respectively. In general, this combined CNN and LSTM model can predict the gene expression data of RNA-seq excellently. The feasibility of the model was verified by multiple replications. The results of multiple replications are shown in Figure S2. The FPKM values obtained from RNA-seq can be used as input to effectively predict their gene expressions subsequently, and valid information can be obtained by this model.

**Statistical analysis using cloud models**

The global gene sequences are modeled and predicted to obtain the residuals of the prediction results. Then, the cloud model is constructed using the residuals as shown in Figure 3A. The cloud drops are constructed by the residuals and the overall distribution is normal. Finally, the outlier identification index is constructed by combining the cloud model theory, and the outlier identification interval is drawn. By using $(Ex-2En, Ex+2En)$ as the confidence interval, the outlier genes beyond the range were found as shown in Figure 3B. The results of multiple replicates are shown in Figure S3. The operator can also adjust the threshold value to $(Ex-En, Ex+En)$ or $(Ex-0.67En, Ex+0.67En)$. However, as the sensitivity increases, the specificity inevitably decreases. Furthermore, operators are not recommended to choose $(Ex-3En, Ex+3En)$ as the recognition interval because its sensitivity will be too low.

**Analysis of Parkinson’s disease-associated genes**

The deferentially expressed genes were obtained by edgeR, DESeq2, and Deep-cloud respectively, as shown in Figure 4A. Deep-cloud could obtain more differential genes, and the differences were statistically significant compared with edgeR and DESeq2. $(P < 0.01)$ In addition, as shown in Figure 4B, the genes obtained by edgeR intersected with those obtained by Deep-cloud, while DESeq2 did not intersect with either of them. By analyzing the GO or KEGG pathways of the differential genes obtained by Deep-cloud, it can be seen that the top pathways include neuron death, regulation of neuron death, regulation of ion trans-membrane transport, anion transport, regulation of membrane potential, cognition, learning or memory, positive regulation of nervous system development, rhythmic process, negative regulation of neuron death, learning, peptide-threonine phosphorylation. (Figure 4CD, Figure S4) The GO pathways obtained from edgeR and DESeq2 are compared in Figure S4. The GO pathways that are accessible to all three methods include regulation of neuronal death, development of central nervous system neurons, learning or memory, negative regulation of movement, etc. The KEGG pathways
common to all three methods include glutamatergic synapses, cAMP signaling pathway, MAPK signaling pathway, TNF signaling pathway, etc. These GO or KEGG pathways have been shown to be closely associated with Parkinson's disease. [24, 25] There are some GO pathways that were only obtained using Deep-cloud analysis of input data, such as dopamine receptor signaling pathways, neuronal cellular homeostasis, leukocyte activation involved in inflammatory responses, intestinal epithelial cell development, regulation of ATP metabolic processes, T cell proliferation, glucose metabolic processes, etc. These pathways were not available with the existing data from edgeR or DESeq2 analysis. In addition, a comparison of the GO pathway network maps obtained by the three methods is shown in Figure S5. Overall, our method presents an additional possibility to analyze RNA-seq data and is able to discover information not obtained by existing methods.

Discussion

Currently, the differential gene analysis approach of RNA-seq does not yet fully exploit the useful information in the data and does not realize the full potential of capturing the biological significance of the transcriptome. It has been demonstrated that tens of thousands of RNA-seq reads are sufficient for differential gene analysis and that higher sequence depth does not improve much. [26] How to perform effective analysis with limited data is an urgent problem at present. Studies have shown that DEseq2 method is the main choice for differential expression analysis in RNA-Seq data. [27] However, for small sample sizes (e.g., sample size < 3), the method has problems in finding deferentially expressed genes. Consequently, deep learning methods were considered to aid in the analysis of RNA-seq data. Several algorithms have been used to solve the problem of feature selection and classification of gene expression data. [28] There is still a need for an accurate and sensitive method for the analysis of RNA-seq gene expression. CNN is one of the most representative algorithms in deep learning, consisting of one or more convolutional and fully-connected layers. CNN is now a fashionable tool for various machine learning applications and a popular tool for research in biomedical fields. More and more CNN models applicable to the biomedical field are emerging. [15] This study combines CNN with LSTM and develops the function of Deep-cloud for evaluation metrics formulation in the biomedical field. The proposed method opens up a new field for the analysis of RNA-seq data. After several iterations of validation, the method can be generally applied to the analysis of gene expression data of RNA-seq.

In this study, the residuals obtained from CNN and LSTM predictions are statistically analyzed by means of a cloud model. The quantitative properties of qualitative concepts are expressed in the cloud model by three numerical features of the cloud, namely $Ex$, $En$ and $He$, which are called cloud expectation, cloud entropy and cloud hyper entropy, respectively. The expectation $Ex$ is derived from probability theory and refers to the expectation of the value of a variable in a random sample. It is a classical quantification of the qualitative concept and describes the statistical characteristics of the data. The entropy $En$, firstly derived from thermodynamics, characterizes the degree of chaos in the state of a material system. It is used in cloud models to combine the ambiguity and probability of qualitative concepts, reflecting the uncertainty of qualitative concepts. It is manifested in two main aspects. On the one hand, it directly reflects the range of values of the cloud drops that can be accepted by the concept in the domain space,
i.e., the vagueness. On the other hand, the higher the entropy, the more vague the concept is, and the more
difficult it is to quantify the certainty. The super entropy $H_e$, the entropy of entropy, is a measure of
uncertainty of entropy. In the cloud model, as the thickness and dispersion of the cloud, which reflects the
dispersion degree of the cloud droplets that make up the cloud model. The greater the super entropy, the
greater the dispersion of the cloud droplets. The greater the randomness of the affiliation, the greater the
thickness of the clouds. Overall, the use of cloud models provides more reliable analysis results for gene
expression in RNA-seq.

In recent years, transcriptome analysis has been of great importance for various diseases, especially
brain research. [29] RNA-seq data from Parkinson's disease and normal controls were used as input in
this study. Most of the current studies on the analysis of the disease mechanism of Parkinson's disease
are based on the differentially expressed genes obtained by RNA-seq. Their results are limited to that step
of differential gene acquisition. Some of the GO or KEGG pathways obtained from the final analysis of
this study are not available from the analysis of other methods. For example, the KEGG pathway unique
to Deep-cloud contains the development of the intestinal epithelium. This is in line with the current
conclusion that Parkinson's disease is associated with intestinal disease. [30] In addition, dopamine plays
an important role in integrin-mediated cellular transport and extravasation of central nervous system and
peripheral T cells. The GO pathway obtained by analyzing the input data by the present method then
contains the dopamine receptor signaling pathway and the regulation of T cell activation. This indicates
that the information obtained by this method is extremely helpful for the study of Parkinson's disease.

Differentially expressed gene prediction and functional screening for RNA-seq are of great significance for
the study of disease pathogenesis, disease detection, disease prevention, drug screening and diagnosis
of complex diseases. The proposed model is of seminal value for the analysis of RNA-seq gene
expression data. In order to fully exploit the information in RNA-seq data, it is necessary to use deep
learning methods. The present method is an excellent attempt. However, there are still some
shortcomings in this study. For example, the total number of data input is 60,000, but there is still the
disadvantage of small amount of data. The amount of experimental data input can be further increased
in the future to obtain a more abundant gene expression analysis result of RNA-seq.

Conclusion

This study focuses on developing a deep neural network based RNA-seq gene expression analysis model,
Deep-cloud. The model utilized CNN and LSTM to construct a deep learning network model excellently to
predict the gene expression of RNA-seq. Its sensitivity is 89.7% and specificity is 83.4%. Then, the
transcriptome data were analyzed more deeply for differential gene expression between disease and
normal groups by integrating the statistical methods of the cloud model. The model further improves the
sensitivity of obtaining differential genes compared to the results obtained with traditional differential
gene analysis software such as DESeq2 and edgeR. It is possible to find more GO or KEGG pathways
related to diseases using this method. More information about differential gene expression of RNA-seq
can be analyzed by this model. Overall, the proposed Deep-cloud opens up a new pathway for the
biomedical field.
Materials And Methods

Acquisition and prepossessing of RNA-seq data

Firstly, RNA-seq data from mice with Parkinson’s disease and normal controls were obtained by the conventional SMART library construction process, which is described in the published paper. [31] The construction of mice for this Parkinson’s disease model has been validated in several articles. [32] Therefore, the true positive rate in this paper is defined as the percentage of genes from the disease group that are predicted to be abnormal by this model. The true negative rate is defined as the proportion of genes from normal controls that are predicted to be normal by the model. The specific data used in this paper can be downloaded from the NCBI website: https://www.ncbi.nlm.nih.gov/bioproject/?term=PRJNA826720. Prior to gene expression analysis, raw data were processed by quality to obtain clean data, mainly including ploy-N read removal, adapter removal and low quality read removal. After obtaining clean data, Hisat2 (v2.0.5) was used to map to the mouse genome to obtain the location information of the reference genome. Finally, the number of reads mapped to each gene was calculated using FeatureCounts. Then the FPKM values of each sample were calculated and normalized by the Fragments Per kilo-base of exon model per Million mapped fragments formula. Data with FPKM values of 0 were excluded and a total of 20,000 transcripts containing both disease and normal groups were randomly selected as input. This was repeated three times for a total of 60,000 transcripts. The specific formula for FPKM was as follows.

\[
FPKM_i = \frac{X_i}{l_iN} \cdot 10^9
\]

In the formula, \(X_i\) is the expression of gene \(i\). \(l_i\) is the sum of all exon lengths of gene \(i\). \(N\) is the expression of all genes in a sample, also called the sequencing depth.

Effective Combination Of Cnn And Lstm

The combined approach of CNN and LSTM can extract complex features from datasets and store complex irregular trends. CNN layers include an input layer that receives variables as input, an output layer that extracts features to the LSTM, and a certain number of hidden layers. [33] The hidden layers usually include a convolutional layer, an activation function, a pooling layer, and a fully connected layer. The CNN layer has enough power to take local features from the higher-level inputs and transfer them to the lower layers to obtain more complex features.[34] Its specific formula is as follows.

\[
y_{ij} = \sigma \left( b_j^1 + \sum_{m=1}^{M} w_{m,j}^1 x_{i+m-1,j}^0 \right)
\]
\[ y_{ij}^c = \sigma \left( b_j^c + \sum_{m=1}^{M} w_{m,j}^c x_{i+m-1,j}^0 \right) \]

\( x \) is the input vector that generates the power, and \( i \) is the number of units in each window. The resultant vector \( y \) of Eq. 2 is the output layer of the first convolution, \( y \) is calculated from the output vector \( x \) of the previous layer, \( b \) represents the bias of the \( j \) feature mapping, \( w \) is the weight of the kernel, \( m \) is the index value of the filter, and \( \sigma \) is the activation function. Eq. 3 is the result of the output of the vector \( y \) of the convolutional layer \( c \).

Next, the maximum pooling layer is used to sub-sample the output of the convolutional layer. Moreover, it is used for two main reasons. First, reduce the computational consumption of the upper layer by eliminating non-extreme values. Second, its ability to extract local dependencies within different regions to maintain the most significant information. The specific formula is as follows.

\[ p_{ij}^c = \max_{r \in R} y_{ix-T+r,j}^{c-1} \]

The maximum pooling layer operation is shown in Eq. 4. \( T \) is the step size that determines the distance to move the input data region and \( R \) is the pooling size and the input size \( y \).

LSTM is a special recurrent neural network (RNN) proposed to solve the deficiency of RNN in the long-range dependence problem. [21] LSTM achieves the selection of new information addition and the control of information accumulation rate by adding a gating mechanism, and the addition of gating control makes up for the deficiency of RNN network structure. Compared with the classical RNN network in which there is only a single \( \tanh \) cyclic body, LSTM adds three gate structures and one memory unit. Let \( W \) be the gate weight and \( b \) be the bias, the gate structure can be described as:

\[ g(x) = \sigma \left( W_x + b \right) \]

The three gates in the LSTM gate structure are the input gate, the output gate and the forget gate. The input gate mainly reflects the number of information stored in the cell state \( c_t \) at the current moment in the input sequence \( x_t \) at this time. The output state of the forget gate mainly determines the amount of information from the cell state \( c_t \) to the output value of the output gate at this moment in time. The gate control state output often uses the \( \text{sigmoid} \) function as the activation function, while the activation functions of the input gates and memory cells usually use \( \tanh \). Each gate state update of the LSTM is calculated as:
\[ i_t = \sigma (W_i x_t + U_i h_{t-1} + b_i) \]

\[ f_t = \sigma (W_f x_t + U_f h_{t-1} + b_f) \]

\[ o_t = \sigma (W_o x_t + U_o h_{t-1} + b_o) \]

Memory unit update:

\[ c_t = \tanh (W_c x_t + U_c h_{t-1} + b_c) \]

Status update:

\[ c_t = f_t \cdot c_{t-1} + i_t \cdot c_t \]

\[ h_t = o_t \cdot \tanh (c_t) \]

The formula \( \sigma \) is the *sigmoid* activation function. \( W_i, W_f, W_o, b_i, b_f, b_o \) are the weight matrices and biases of the input gate, forgetting gate and output gate. \( W_c, b_c \) are the weight matrices and biases of the memory cell after updating. \( U_i, U_f, U_o, U_c \) are the state quantity weight matrices. \( h_{t-1} \) is the previous moment state quantity.

The last layer of CNN and LSTM is composed of fully connected layers. The final output of the LSTM unit is the feature vector \( h_c \), where \( c \) represents the number of units in the LSTM.

\[ h_c = [h_1, h_2, \ldots, h_c] \]

The equation of the last step is expressed as follows:
\[ d_i^c = \sum_j w_{ji}^{c-1} \left[ \sigma(h_i^{c-1}) + b_i^{c-1} \right] \]

\( w \) is the weight of node \( i \) at layer \( c - 1 \) and node \( j \) at layer \( c \). \( b_i^{c-1} \) denotes the bias amount.

**Statistical Methods Of Cloud Models**

**Basic concepts of cloud model**

Cloud model is a cognitive model based on probabilistic statistics and fuzzy set theory to achieve bidirectional cognitive transformation between qualitative concepts and quantitative data. [22] It has been widely studied and applied to intelligent control, data mining, system evaluation, qualitative assessment, etc. A forward normal cloud transformation algorithm automatically generates sample data named cloud drops by two normal random number generators based on three features (\( Ex, En \) and \( He \)). The overall shape of the cloud responds to a qualitative concept expressed in quantitative values. The cloud model uses the three numerical characteristics \( Ex \) (expectation), \( En \) (entropy) and \( He \) (super-entropy) to portray the connotation of a concept, denoted as \( C(Ex, En, He) \). This is also the feature vector represented by the numerical features in the cloud model, i.e., the cloud vector. The digital features of clouds lie in the fact that a certain number of cloud drops of a certain size can be determined with \( Ex, En \) and \( He \) to form clouds and integrate the fuzziness and randomness of the concept. The cloud models constructed when \( Ex, En \) and \( He \) take different values exhibit different cloud shapes.

We assume that \( U \) is a quantitative domain represented by numerical values and \( T \) is a qualitative representation of a concept on \( U \). If a quantitative value \( x \in U \) is a sequential consequent realization of a qualitative concept \( T \), then the affiliation of \( x \) to \( T \) is denoted as \( \mu(x) \in [0,1] \), and \( \mu(x) \) is a random function with a certain probability distribution. The distribution of \( x \) over the theoretical domain \( U \) is called a cloud, i.e., a cloud model of the concept \( T \), and \( x \) is called a cloud drop. From the definition of the cloud model, it is clear that the affiliation \( \mu(x) \) of a point \( x \) on the theoretical domain \( U \) is not fixed, but presents a certain randomness and ambiguity of subtle changes. However, these changes do not affect the overall characteristics of the cloud. Therefore, the cloud model is viewed as a cloud with no clear boundary, and numerous cloud drops constitute the cloud. However, the affiliation degree of cloud drops is not fixed again. The detailed schematic is shown in Fig. 5.

**Forward Cloud Generator**

Cloud droplets appear based on the numerical characteristics of clouds. When their number reaches a certain level they gather into clouds, and this process is realized by the forward cloud generator. The forward cloud generator is a mapping from qualitative concepts to quantitative data, but the mapping is
a non-fixed transformation model. The forward cloud generator completes the conversion of qualitative concepts into quantitative data and describes the distribution process, which is forward and directive. Its whole process is shown in Fig. 6A.

The input of the forward-normal cloud generator \( CG(E_x', E_n', H_e', n) \) algorithm is the numerical features \( (E_x', E_n', H_e') \) representing the qualitative concept and the number of generated cloud drops \( n \). The output is the quantitative cloud with affiliation \( \mu \) (i.e., \( drop(x_i, \mu_i), i = 1, 2, ..., n \)), and \( x \) is the cloud drop. The specific algorithm steps are as follows.

Step 1: Generate a normal random number \( E_{n}' \), where the expectation is \( E_n \) and the variance is \( H_e \).

Step 2: Generate a normal random number \( x \), where the expectation is \( E_x \) and the variance is \( E_{n}' \).

Step 3: Let \( x \) be a certain quantize value of the qualitative linguistic value concept 1, i.e., a cloud drop.

Step 4: The formula is given by.

\[
y = \exp \left[ \frac{-(x - E_x)^2}{2E_{n}'} \right]
\]

Step 5: Let \( y \) be the affiliation of \( x \) to this qualitative linguistic value concept.

Step 6: \( \{x, y\} \) a complete representation of this transformation between qualitative and quantitative.

Step 7: Repeat steps 1–6 until a sufficient number of cloud drops are generated.

**Reverse Cloud Generator**

The model that realizes the conversion from quantitative statistics to qualitative concepts is called the inverse cloud generator, which is the opposite of the forward cloud generator. It can transform quantitative and accurate data into qualitative concepts expressed by numerical features, and through these numerical features react to the entire morphology of the cloud droplets represented by the previous data. The more the amount of data used to describe the cloud droplets, the more accurate the concepts it has to transform. This approach to cloud generation is a reverse, indirect process, the entire flow of which is shown in Fig. 6B.

The principle of the inverse cloud generator is mathematical statistics, and its algorithm is divided into two types based on the presence or absence of affiliation information: affiliation-based and affiliation-
The data of RNA-seq are affiliation-free. The affiliation-free inverse cloud generator algorithm takes a set of cloud drops as the input sample, where the distribution of the drops is required to be regular. Then, the generator takes the numerical features as the output and uses them to describe the concept of qualitative linguistic values. The input of this algorithm is each sample point \( x_i, i = 1, 2, \ldots, n \). The output is the numerical feature values \((E_x, E_n, H_e)\), and its specific steps are as follows.

**Step 1:** Calculate the sample \( x_i \) mean value.

\[
\bar{X} = \frac{1}{n} \sum_{i=1}^{n} x_i
\]

**15**

First-order sample absolute center distance:

\[
\frac{1}{n} \sum_{i=1}^{n} \left| x_i - \bar{X} \right|
\]

**16**

Sample variance:

\[
S^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{X})^2
\]

**17**

**Step 2:** Calculate the mathematical expectation of the cloud model:

\[E_x = \bar{X}\] (18)

**Step 3:** Calculate the entropy of the cloud model:

\[E_n = \sqrt{\frac{n}{2}} \times \frac{1}{n} \sum_{i=1}^{n} \left| x_i - E_x \right|\] (19)

**Step 4:** Calculate the super entropy of the cloud model:

\[H_e = \sqrt{S^2 - E_n^2}\]
Subsequent Analysis

The ROC and corresponding area under the curve were obtained from the pROC package. Gene ontology enrichment and KEGG analysis of deferentially expressed genes were implemented by the cluster Profiler R package.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent to publish

Manuscript is approved by all authors for publication.

Availability of data and materials

The data used in this paper can be downloaded from the NCBI website: https://www.ncbi.nlm.nih.gov/bioproject/?term=PRJNA826720.

The source code used to support the findings of this study are freely available to the academic community at https://github.com/habbyzy/Deep-cloud.

Conflicts of Interest

The authors declare no conflict of interest.

Funding

This research was supported by the National Natural Science Foundation of China [61801108] and Natural Science Foundation of Jiangsu Province [BK20201148, BK20211166]. The funding agency provided the funding for the manuscript.

Authors’ contributions

Author contributions: Y. Z. did the investigation, formal analysis, writing original draft, data analysis. E.J. did the investigation, methodology. H.S. did the investigation, methodology. Z.L. did the investigation, methodology. Y.S. did the data analysis. M.P. did the investigation, methodology. J.T. did the supervision, resources. Q.G. did the methodology, conceptualization, writing review and editing. Z.L. did the supervision. All authors have read and approved the final manuscript.

Acknowledgments

Not applicable.
References


Figures

**Figure 1**

**General flow chart of Deep-cloud.** First, a deep learning network model is constructed using CNN and LSTM to predict the gene expression of RNA-seq. Then, the predicted model is saved and the cloud model
is drawn using its residuals. Finally, the differential gene expression beyond the evaluation interval is analyzed and obtained by combining the statistical methods of the cloud model.

Figure 2

Prediction results of the combination of CNN and LSTM.

A: It is the iteration curves of the corresponding loss function for the training and validation sets. It tends to be stable at 100 iterations, and the model has strong robustness and generalization ability. B: It is the ROC curve and AUC value of CNN and LSTM model. Its AUC value is 0.93, which indicates that the method has high prediction accuracy.

Figure 3
Cloud model and subsequent analysis.

A: Cloud model was constructed using predicted residuals. B: Outlier identification indexes were constructed by combining cloud model theory, and outlier identification intervals were plotted to find outlier genes beyond the range.

Figure 4

Comparison of differential expressed genes and GO or KEGG analysis.

A: Differential expressed genes were obtained by edgeR, DESeq2 and Deep-cloud. Deep-cloud could obtain more differential expressed genes, and the differences were statistically significant compared with edgeR and DESeq2. (P < 0.01) B: The genes obtained by edgeR and Deep-cloud intersected, while DESeq2 did not intersect genes with either of them. C: GO pathway obtained by Deep-cloud from differential expressed gene analysis. D: KEGG pathway obtained by Deep-cloud from differential expressed gene analysis.
Figure 5

The diagram of the construction of Cloud model.
Figure 6

Schematic diagram of the forward (A) and reverse (B) cloud generators.