R documentation

of 'prepareDataForTraining.Rd'

September 3, 2020

`prepareDataForTraining` *Prepare training and test final data for training and evaluation Deep Neural Network model.*

Description

Prepare training and test final data for training and evaluating Deep Neural Network model. Expression matrix is normalized by CPMs (counts per million) in log2-space and normalized. Samples are shuffled in order to avoid biases during training. Note that expression matrix is transposed in order to prepare data for training.

Usage

```
prepareDataForTraining(
  object,
  type.data,
  combine = "both",
  file.backend = NULL,
  number.rows = NULL,
  compression.level = NULL,
  verbose = TRUE
)
```

Arguments

`object`	`DigitalDLSorter` object with `single.cell.final` and `prob.cell.types` slots.
`type.data`	Type of data to generate among 'train', 'test' or 'both' (the last by default).
`combine`	Character determining if combine training data. Can be 'both', 'bulk' or 'single-cell' ('both' by default). Note that test data is always combined.
`file.backend`	A valid file path where to save the HDF5 file used as back-end. If it is equal to `NULL` (by default), the data are loaded in memory.

number.rows — HDF5 file is saved by row chunks in order to improve the execution times during training. This is because trainDigitalDLSorterModel only access to data by rows (samples). You can provided the number of rows that are stored together in each chunk. Note that the more columns the more RAM is used, although execution times are improved.

verbose — Show informative messages during the execution.

Details

This function allows you to select which kind of data you want to use for training: single-cell profiles, bulk profiles or a combination of both. See combine argument for details. We recommend the use of the combination or the bulk profiles, since the results are better. For test data, profiles are combined in any case, but during the evaluation of results you can filter single-cell profiles (see calculateEvalMetrics).

digitalDLSorteR allows the use of HDF5 files as back-end for the resulting data using DelayedArray and HDF5Array packages in cases of generating too large expression matrix. This functionality allows you to work without keeping the data loaded in memory, which will be of vital importance during some computationally heavy steps such as neural network training. You must provide a valid file path in file.backend argument to store the resulting file with '.h5' extension. The data will be accessible from R without being loaded into memory. This option slightly slows down execution times, since subsequent transformations of data will be carried out by chunks instead of using all data. We recommend this option due to the large size of the simulated matrices.

Value

A DigitalDLSorter object with final.data slot containing a list with one or two entries (depending on selected type.data argument): 'train' and 'test'. Each entry contains a SummarizedExperiment object with single-cell and bulk samples combined in assay slot, sample names in rowData slot and feature names in colData slot.

See Also

generateBulkSamples generateTrainAndTestBulkProbMatrix

Examples

```
## loading all data in memory
DDLSChung <- prepareDataForTraining(
  object = DDLSChung,
  type.data = "both",
  verbose = TRUE
)
## Not run:
## using HDF5 as backend
DDLSChung <- prepareDataForTraining(
  object = DDLSChung,
  type.data = "both",
  combine = "both",
  file.backend = "DDLSChung.final.data.combined.h5",
  verbose = TRUE
)

## End(Not run)
```

Index