Package ‘janus’

December 16, 2022

Type Package
Title Optimized Recommending System Based on 'tensorflow'
Version 1.0.0
Maintainer Giancarlo Vercellino <giancarlo.vercellino@gmail.com>
Description Proposes a coarse-to-fine optimization of a recommending system based on deep-neural networks using 'tensorflow'.
License GPL-3
Encoding UTF-8
RoxygenNote 7.2.1
Imports keras (>= 2.9.0), tensorflow (>= 2.9.0), dplyr (>= 1.0.10), purrr (>= 0.3.4), forcats (>= 0.5.1), tictoc (>= 1.0.1), readr (>= 2.1.2), ggplot2 (>= 3.3.6), narray (>= 0.4.1.1), lubridate (>= 1.7.10), RcppAlgos (>= 2.6.0), Rmpfr (>= 0.8-7), Metrics (>= 0.1.4), StatRank (>= 0.0.6), hash (>= 2.2.6.2), reticulate (>= 1.26)
URL https://rpubs.com/giancarlo_vercellino/janus
Suggests testthat (>= 3.0.0)
Config/testthat/edition 3
NeedsCompilation no
Author Giancarlo Vercellino [aut, cre, cph]
Repository CRAN
Date/Publication 2022-12-16 10:00:02 UTC

R topics documented:

janus ................................................. 2

Index 5
Description

Coarse-to-fine optimization of a recommending system based on deep neural networks with Tensorflow/Keras back-end

Usage

```r
janus(
  data,
  rating_label,
  rater_label,
  rated_label,
  task,
  skipShortcut = FALSE,
  rater_embedding_size = c(8, 32),
  rated_embedding_size = c(8, 32),
  layers = c(1, 5),
  activations = c("relu", "selu", "relu", "sigmoid", "softmax", "softplus", "softsign",
               "tanh", "linear", "leaky_relu", "parametric_relu", "thresholded_relu", "swish",
               "gelu", "mish", "bent"),
  nodes = c(8, 512),
  regularization_L1 = c(0, 100),
  regularization_L2 = c(0, 100),
  dropout = c(0, 1),
  batch_size = 64,
  epochs = 10,
  optimizer = c("adam", "sgd", "adamax", "adagrad", "nadam", "rmsprop"),
  opt_metric = "bac",
  folds = 3,
  reps = 1,
  holdout = 0.1,
  n_steps = 3,
  n_samp = 10,
  offset = 0,
  n_top = 3,
  seed = 999,
  verbose = TRUE
)
```

Arguments

- **data**: A data frame including at least three features: rating actor, rated item and rating value.
- **rating_label**: String. Single label for the feature containing the rating values.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rater_label</td>
<td>String. Single label for the feature containing the rating actors.</td>
</tr>
<tr>
<td>rated_label</td>
<td>String. Single label for the feature containing the rated items.</td>
</tr>
<tr>
<td>task</td>
<td>String. Available options are: &quot;regr&quot;, for regression (when the rating value is numeric); &quot;classif&quot;, for classification (when the rating value is a class or a factor).</td>
</tr>
<tr>
<td>skipShortcut</td>
<td>Logical. Option to add a skip shortcut to improve network performance in case of many layers. Default: FALSE.</td>
</tr>
<tr>
<td>rater_embedding_size</td>
<td>Integer. Output dimension for embedding the rating actors. Default: coarse-to-fine search (8 to 32).</td>
</tr>
<tr>
<td>rated_embedding_size</td>
<td>Integer. Output dimension for embedding the rated items. Default: coarse-to-fine search (8 to 32).</td>
</tr>
<tr>
<td>layers</td>
<td>Positive integer. Number of layers for DNN. Default: coarse-to-fine search (1 to 5).</td>
</tr>
<tr>
<td>nodes</td>
<td>Positive integer. Integer vector with nodes for each layer. Default: coarse-to-fine search (8 to 512).</td>
</tr>
<tr>
<td>regularization_L1</td>
<td>Positive numeric. Value for L1 regularization of loss function. Default: coarse-to-fine search (0 to 100).</td>
</tr>
<tr>
<td>regularization_L2</td>
<td>Positive numeric. Value for L2 regularization of loss function. Default: coarse-to-fine search (0 to 100).</td>
</tr>
<tr>
<td>dropout</td>
<td>Positive numeric. Value for dropout parameter at each layer (bounded between 0 and 1). Default: coarse-to-fine search (0 to 1).</td>
</tr>
<tr>
<td>batch_size</td>
<td>Positive integer. Maximum batch size for training. Default: 64.</td>
</tr>
<tr>
<td>folds</td>
<td>Positive integer. Number of folds for repeated cross-validation. Default: 3.</td>
</tr>
<tr>
<td>holdout</td>
<td>Positive numeric. Percentage of cases for holdout validation. Default: 0.1.</td>
</tr>
<tr>
<td>n_steps</td>
<td>Positive integer. Number of phases for the coarse-to-fine optimization process (minimum 2). Default: 3.</td>
</tr>
</tbody>
</table>

offset Positive numeric. Percentage of expansion of numeric boundaries during the coarse-to-fine optimization. Default: 0.

n_top Positive integer. Number of candidates selected during the coarse-to-fine phase. Default: 3.

seed Positive integer. Seed value to control random processes. Default: 42.

verbose Positive integer. Seed value to control random processes. Default: TRUE.

Value

This function returns a list including:

- pipeline:
  - model:
    - configuration: DNN hyper-parameters (layers, activations, regularization_L1, regularization_L2, nodes, dropout)
    - model: Keras standard model description
    - recommend: function to use to recommend on rating actors
    - plot: Keras standard history plot
    - training_metrics: tracking of opt_metric across folds and repetitions
    - test_frame: testing set with the related predictions, including
      - testing_metrics: summary statistics for testing
  - time_log

Author(s)

Maintainer: Giancarlo Vercellino <giancarlo.vercellino@gmail.com> [copyright holder]
Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

See Also

Useful links:

- https://rpubs.com/giancarlo_vercellino/janus
Index

janus, 2
janus-package (janus), 2