Package: spinner (via r-universe)

November 24, 2024

Type Package

Title An Implementation of Graph Net Architecture Based on 'torch'

Version 1.1.0

Maintainer Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

Description Proposes a 'torch' implementation of Graph Net architecture allowing different options for message passing and feature embedding.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.1

Imports torch (>= 0.9.0), igraph (>= 1.3.5), purrr (>= 0.3.4), ggplot2 (>= 3.3.6), ggthemes (>= 4.2.4), tictoc (>= 1.0.1), readr (>= 2.1.2), lubridate (>= 1.7.10), rlist (>= 0.4.6.2), fastDummies (>= 1.6.3), entropy (>= 1.3.1), abind (>= 1.4-5)

URL https://rpubs.com/giancarlo_vercellino/spinner

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Giancarlo Vercellino [aut, cre, cph]

Date/Publication 2023-03-21 06:10:02 UTC

Config/pak/sysreqs libglpk-dev libicu-dev libxml2-dev libx11-dev

Repository https://pigian.r-universe.dev

RemoteUrl https://github.com/cran/spinner

RemoteRef HEAD

RemoteSha 1331eb5e8b7c417651978df0609241924ac5c4f6

Contents

spinner *spinner*

Description

Spinner is an implementation of Graph Nets based on torch. Graph Nets are a family of neural network architectures designed for processing graphs and other structured data. They consist of a set of message-passing operations, which propagate information between nodes and edges in the graph, and a set of update functions, which compute new node and edge features based on the received messages.

Proposes a 'torch' implementation of Graph Net architecture allowing different options for message passing and feature embedding.

Usage

```
spinner(
  graph,
  target,
  node_labels = NA,
  edge_labels = NA,
  context_labels = NA,
  direction = "from_head",
  sampling = NA,
  threshold = 0.01,
  method = "null",
  node_embedding_size = 5,
  edge_embedding_size = 5,
  context_embedding_size = 5,
  update_order = "enc",
  n_layers = 3,
  skip_shortcut = FALSE,
  forward_layer = 32,
  forward_activation = "relu",
  forward_drop = 0.3,
  mode = "sum",
  optimization = "adam",
  epochs = 100,
  lr = 0.01,
  patience = 30,
  weight_decay = 0.001,
  reps = 1,
  folds = 3,
  holdout = 0.2,
  verbose = TRUE,
  seed = 42
)
```

Arguments

graph	A graph in igraph format (without name index for nodes).
target	String. Predicted dimension. Options are: "node", "edge".
node_labels	String. Character vector with labels of node features. In case of absent features, default to NA (automatic node embedding with selected method).
edge_labels	String. Character vector with labels of edge features. In case of absent features, default to NA (automatic edge embedding with selected method).
context_labels	String. Character vector with labels of context features. In case of absent features, default to NA (automatic context embedding with selected method).
direction	String. Direction of message propagation. Options are: "from_head", "from_tail". Default to: "from_head".
sampling	Positive numeric or integer. In case of huge graph, you can opt for a subgraph. Sampling dimension expressed in absolute value or percentage. Default: NA (no sampling).
threshold	Numeric. Below this threshold (calculated on edge density), sampling is done on edges, otherwise on nodes. Default: 0.01.
method	String. Embedding method in case of absent features. Options are: "null" (zeroed tensor), "laplacian", "adjacency". Default: "null".
node_embedding_size	Integer. Size for node embedding. Default: 5.
edge_embedding_size	Integer. Size for edge embedding. Default: 5.
context_embedding_size	Integer. Size for node embedding. Default: 5.
update_order	String. The order of message passing through nodes (n), edges (e) and context (c) for updating information. Available options are: "enc", "nec", "cen", "ecn", "nce", "cne". Default: "enc".
n_layers	Integer. Number of graph net variant layers. Default: 1.
skip_shortcut	Logical. Flag for applying skip shortcut after the graph net variant layers. Default: FALSE.
forward_layer	Integer. Single integer vector with size for forward net layer. Default: 32 (layers with 32 nodes).
forward_activation	String. Single character vector with activation for forward net layer. Available options are: "linear", "relu", "mish", "leaky_relu", "celu", "elu", "gelu", "selu", "bent", "softmax", "softmin", "softsign", "sigmoid", "tanh". Default: "relu".
forward_drop	Numeric. Single numeric vector with drop out for forward net layer. Default: 0.3.
mode	String. Aggregation method for message passing. Options are: "sum", "mean", "max". Default: "sum".
optimization	String. Optimization method. Options are: "adadelta", "adagrad", "rmsprop", "rprop", "sgd", "asgd", "adam".

epochs	Positive integer. Default: 100.
lr	Positive numeric. Learning rate. Default: 0.01.
patience	Positive integer. Waiting time (in epochs) before evaluating the overfit performance. Default: 30.
weight_decay	Positive numeric. L2-Regularization weight. Default: 0.001.
reps	Positive integer. Number of repeated measures. Default: 1.
folds	Positive integer. Number of folds for each repetition. Default: 3.
holdout	Positive numeric. Percentage of nodes for testing (edges are computed accordingly). Default: 0.2.
verbose	Logical. Default: TRUE
seed	Random seed. Default: 42.

Value

This function returns a list including:

- graph: analyzed graph is returned (original graph or sampled subgraph).
- model_description: general model description.
- model_summary: summary for each torch module.
- pred_fun: function to predict on new graph data (you need to add new nodes/edges to the original graph respecting the directionality).
- cv_error: cross-validation error for each repetition and each fold. The error is a weighted normalized loss based on mse and binary cross-entropy (depending on the nature of each specific feature).
- summary_errors: final summary of error during cross-validation and testing.
- history: plot with loss for final training and testing.
- time_log: computation time.

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

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

See Also

Useful links:

- https://rpubs.com/giancarlo_vercellino/spinner

spinner_random_search *spinner_random_search*

Description

spinner_random_search is a function for fine-tuning using random search on the hyper-parameter space of spinner (predefined or custom).

Usage

```
spinner_random_search(
  n_samp,
  graph,
  target,
  node_labels = NA,
  edge_labels = NA,
  context_labels = NA,
  direction = NULL,
  sampling = NA,
  threshold = 0.01,
  method = NULL,
  node_embedding_size = NULL,
  edge_embedding_size = NULL,
  context_embedding_size = NULL,
  update_order = NULL,
  n_layers = NULL,
  skip_shortcut = NULL,
  forward_layer = NULL,
  forward_activation = NULL,
  forward_drop = NULL,
  mode = NULL,
  optimization = NULL,
  epochs = 100,
  lr = NULL,
  patience = 30,
  weight_decay = NULL,
  reps = 1,
  folds = 2,
  holdout = 0.2,
  verbose = TRUE,
  seed = 42,
  keep = FALSE
)
```

Arguments

n_samp	Positive integer. Number of models to be randomly generated sampling the hyper-parameter space.

graph	A graph in igraph format (without name index for nodes).
target	String. Predicted dimension. Options are: "node", "edge".
node_labels	String. Character vector with labels of node features. In case of absent features, default to NA (automatic node embedding with selected method).
edge_labels	String. Character vector with labels of edge features. In case of absent features, default to NA (automatic edge embedding with selected method).
context_labels	String. Character vector with labels of context features. In case of absent features, default to NA (automatic context embedding with selected method).
direction	String. Direction of message propagation. Options are: "from_head", "from_tail". Default to: "from_head".
sampling	Positive numeric or integer. In case of huge graph, you can opt for a subgraph. Sampling dimension expressed in absolute value or percentage. Default: NA (no sampling).
threshold	Numeric. Below this threshold (calculated on edge density), sampling is done on edges, otherwise on nodes. Default: 0.01.
method	String. Embedding method in case of absent features. Options are: "null" (zeroed tensor), "laplacian", "adjacency". Default: "null".
node_embedding_size	Integer. Size for node embedding. Default: 5.
edge_embedding_size	Integer. Size for edge embedding. Default: 5.
context_embedding_size	Integer. Size for node embedding. Default: 5.
update_order	String. The order of message passing through nodes (n), edges (e) and context (c) for updating information. Available options are: "enc", "nec", "cen", "ecn", "nce", "cne". Default: "enc".
n_layers	Integer. Number of graph net variant layers. Default: 1.
skip_shortcut	Logical. Flag for applying skip shortcut after the graph net variant layers. Default: FALSE.
forward_layer	Integer. Single integer vector with size for forward net layer. Default: 32 (layers with 32 nodes).
forward_activation	String. Single character vector with activation for forward net layer. Available options are: "linear", "relu", "mish", "leaky_relu", "celu", "elu", "gelu", "selu", "bent", "softmax", "softmin", "softsign", "sigmoid", "tanh". Default: "relu".
forward_drop	Numeric. Single numeric vector with drop out for forward net layer. Default: 0.3.
mode	String. Aggregation method for message passing. Options are: "sum", "mean", "max". Default: "sum".
optimization	String. Optimization method. Options are: "adadelta", "adagrad", "rmsprop", "rprop", "sgd", "asgd", "adam".
epochs	Positive integer. Default: 100.
lr	Positive numeric. Learning rate. Default: 0.01.

patience	Positive integer. Waiting time (in epochs) before evaluating the overfit performance. Default: 30.
weight_decay	Positive numeric. L2-Regularization weight. Default: 0.001.
reps	Positive integer. Number of repeated measures. Default: 1.
folds	Positive integer. Number of folds for each repetition. Default: 3.
holdout	Positive numeric. Percentage of nodes for testing (edges are computed accordingly). Default: 0.2.
verbose	Logical. Default: TRUE
seed	Random seed. Default: 42.
keep	Logical. Flag to TRUE to keep all the explored models. Default: FALSE.

Value

This function returns a list including:

- random_search: summary of the sampled hyper-parameters and average error metrics.
- best: best model according to overall ranking on all average error metrics (for negative metrics, absolute value is considered).
- time_log: computation time.
- all_models: list with all generated models (if keep flagged to TRUE).

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

References

https://rpubs.com/giancarlo_vercellino/spinner

Index