Package ‘tfhub’

Type Package
Title Interface to 'TensorFlow' Hub
Version 0.8.1
Description 'TensorFlow' Hub is a library for the publication, discovery, and consumption of reusable parts of machine learning models. A module is a self-contained piece of a 'TensorFlow' graph, along with its weights and assets, that can be reused across different tasks in a process known as transfer learning. Transfer learning train a model with a smaller dataset, improve generalization, and speed up training.
License Apache License 2.0
URL https://github.com/rstudio/tfhub
BugReports https://github.com/rstudio/tfhub/issues
SystemRequirements TensorFlow >= 2.0 (https://www.tensorflow.org/)
Encoding UTF-8
RoxygenNote 7.1.2
Imports reticulate (>= 1.9.0.9002), tensorflow (>= 1.8.0.9006), magrittr, rstudioapi (>= 0.7), vctrs
Suggests testthat (>= 2.1.0), knitr, tfestimators, keras, rmarkdown, callr, recipes, tibble, abind, fs,
VignetteBuilder knitr
Config/reticulate list( packages = list( list(package = "tensorflow_hub", pip = TRUE) )
NeedsCompilation no
Author Tomasz Kalinowski [aut, cre], Daniel Falbel [aut], JJ Allaire [aut], RStudio [cph, fnd], Google Inc. [cph]
Maintainer Tomasz Kalinowski <tomasz.kalinowski@rstudio.com>
Repository CRAN
Date/Publication 2021-12-19 01:50:02 UTC
R topics documented:

- bake.step_pretrained_text_embedding .............................................. 2
- hub_image_embedding_column ...................................................... 2
- hub_load ................................................................. 3
- hub_sparse_text_embedding_column .................................................. 4
- hub_text_embedding_column .......................................................... 5
- install_tffhub ................................................................. 5
- layer_hub ................................................................. 6
- prep.step_pretrained_text_embedding ............................................... 7
- step_pretrained_text_embedding .................................................. 7

Index 9

---

bake.step_pretrained_text_embedding

_Bake method for step_pretrained_text_embedding_

**Description**

Bake method for step_pretrained_text_embedding

**Usage**

bake.step_pretrained_text_embedding(object, new_data, ...)

**Arguments**

- object : object
- new_data : new data to apply transformations
- ... : One or more selector functions to choose variables.

---

hub_image_embedding_column

_Module to construct a dense 1-D representation from the pixels of images._

**Description**

Module to construct a dense 1-D representation from the pixels of images.

**Usage**

hub_image_embedding_column(key, module_spec)
**Arguments**

- **key**  
  A string or [feature_column](https://tensorflow.rstudio.com/tfevaluators/articles/feature_columns.html) identifying the text feature.

- **module_spec**  
  A string handle or a `_ModuleSpec` identifying the module.

**Details**

This feature column can be used on images, represented as float32 tensors of RGB pixel data in the range [0,1].

---

**hub_load**  

**Hub Load**

---

**Description**

Loads a module from a handle.

**Usage**

```r
hub_load(handle, tags = NULL)
```

**Arguments**

- **handle**  
  (string) the Module handle to resolve.

- **tags**  
  A set of strings specifying the graph variant to use, if loading from a v1 module.

**Details**

Currently this method is fully supported only with Tensorflow 2.x and with modules created by calling `export_savedmodel`. The method works in both eager and graph modes.

Depending on the type of handle used, the call may involve downloading a TensorFlow Hub module to a local cache location specified by the `TFHUB_CACHE_DIR` environment variable. If a copy of the module is already present in the `TFHUB_CACHE_DIR`, the download step is skipped.

Currently, three types of module handles are supported: 1) Smart URL resolvers such as tfhub.dev, e.g.: https://tfhub.dev/google/nlm-en-dim128/1. 2) A directory on a file system supported by Tensorflow containing module files. This may include a local directory (e.g. `/usr/local/mymodule`) or a Google Cloud Storage bucket (gs://mymodule). 3) A URL pointing to a TGZ archive of a module, e.g. https://example.com/mymodule.tar.gz.
hub_sparse_text_embedding_column

Examples

```r
## Not run:
model <- hub_load('https://tfhub.dev/google/tf2-preview/mobilenet_v2/feature_vector/4')

## End(Not run)
```

---

**hub_sparse_text_embedding_column**

*Module to construct dense representations from sparse text features.*

**Description**

The input to this feature column is a batch of multiple strings with arbitrary size, assuming the input is a SparseTensor.

**Usage**

```r
hub_sparse_text_embedding_column(
  key,  # A string or [feature_column](https://tensorflow.rstudio.com/tfestimators/articles/feature_columns.html) identifying the text feature.
  module_spec,  # A string handle or a _ModuleSpec identifying the module.
  combiner,  # a string specifying reducing op for embeddings in the same Example. Currently, 'mean', 'sqrtm', 'sum' are supported. Using 'combiner = NULL' is undefined.
  default_value,  # default value for Examples where the text feature is empty. Note, it’s recommended to have default_value consistent OOV tokens, in case there was special handling of OOV in the text module. If 'NULL', the text feature is assumed be non-empty for each Example.
  trainable = FALSE  # Whether or not the Module is trainable. ‘FALSE’ by default, meaning the pre-trained weights are frozen. This is different from the ordinary ‘tf.feature_column.embedding_column()', but that one is intended for training from scratch.
)
```

**Arguments**

- **key**: A string or [feature_column](https://tensorflow.rstudio.com/tfestimators/articles/feature_columns.html) identifying the text feature.
- **module_spec**: A string handle or a _ModuleSpec identifying the module.
- **combiner**: A string specifying reducing op for embeddings in the same Example. Currently, 'mean', 'sqrtm', 'sum' are supported. Using 'combiner = NULL' is undefined.
- **default_value**: A default value for Examples where the text feature is empty. Note, it’s recommended to have default_value consistent OOV tokens, in case there was special handling of OOV in the text module. If 'NULL', the text feature is assumed be non-empty for each Example.
- **trainable**: Whether or not the Module is trainable. ‘FALSE’ by default, meaning the pre-trained weights are frozen. This is different from the ordinary ‘tf.feature_column.embedding_column()’, but that one is intended for training from scratch.
**hub_text_embedding_column**

**Details**

This type of feature column is typically suited for modules that operate on pre-tokenized text to produce token level embeddings which are combined with the combiner into a text embedding. The combiner always treats the tokens as a bag of words rather than a sequence.

The output (i.e., transformed input layer) is a DenseTensor, with shape `[batch_size, num_embedding_dim]`.

---

**Module to construct a dense representation from a text feature.**

---

**Description**

This feature column can be used on an input feature whose values are strings of arbitrary size.

**Usage**

```
hub_text_embedding_column(key, module_spec, trainable = FALSE)
```

**Arguments**

- **key**
  A string or [feature_column](https://tensorflow.rstudio.com/tfestimators/articles/feature_columns.html) identifying the text feature.

- **module_spec**
  A string handle or a `_ModuleSpec` identifying the module.

- **trainable**
  Whether or not the Module is trainable. ‘FALSE’ by default, meaning the pre-trained weights are frozen. This is different from the ordinary ‘tf.feature_column.embedding_column()’, but that one is intended for training from scratch.

---

**install_tfhub**

**Install TensorFlow Hub**

---

**Description**

This function is used to install the TensorFlow Hub python module.

**Usage**

```
install_tfhub(version = "release", ..., restart_session = TRUE)
```

**Arguments**

- **version**
  version of TensorFlow Hub to be installed.

- **...**
  other arguments passed to [reticulate::py_install()].

- **restart_session**
  Restart R session after installing (note this will only occur within RStudio).
Description

Wraps a Hub module (or a similar callable) for TF2 as a Keras Layer.

Usage

layer_hub(object, handle, trainable = FALSE, arguments = NULL, ...)

Arguments

object    Model or layer object
handle    a callable object (subject to the conventions above), or a string for which `hub_load()`
          returns such a callable. A string is required to save the Keras config of this Layer.
trainable Boolean controlling whether this layer is trainable.
arguments optionally, a list with additional keyword arguments passed to the callable. These
             must be JSON-serializable to save the Keras config of this layer.
...       Other arguments that are passed to the TensorFlow Hub module.

Details

This layer wraps a callable object for use as a Keras layer. The callable object can be passed directly,
or be specified by a string with a handle that gets passed to `hub_load()`.

The callable object is expected to follow the conventions detailed below. (These are met by TF2-
compatible modules loaded from TensorFlow Hub.)

The callable is invoked with a single positional argument set to one tensor or a list of tensors
containing the inputs to the layer. If the callable accepts a training argument, a boolean is passed
for it. It is ‘TRUE’ if this layer is marked trainable and called for training.

If present, the following attributes of callable are understood to have special meanings: variables:
a list of all tf.Variable objects that the callable depends on. trainable_variables: those elements of
variables that are reported as trainable variables of this Keras Layer when the layer is trainable.
regularization_losses: a list of callables to be added as losses of this Keras Layer when the layer is
trainable. Each one must accept zero arguments and return a scalar tensor.

Examples

## Not run:

library(keras)

model <- keras_model_sequential() %>%
  layer_hub(
    handle = "https://tfhub.dev/google/tf2-preview/mobilenet_v2/feature_vector/4",
)

layer_hub
prep.step_pretrained_text_embedding

```r
input_shape = c(224, 224, 3)
)
layer_dense(1)
```

## End(Not run)

---

prep.step_pretrained_text_embedding

Prep method for step.pretrained_text_embedding

**Description**

Prep method for step.pretrained_text_embedding

**Usage**

`prep.step_pretrained_text_embedding(x, training, info = NULL, ...)`

**Arguments**

- `x`: object
- `training`: whether or not it’s training
- `info`: variables state
- `...`: One or more selector functions to choose variables.

---

step_pretrained_text_embedding

Pretrained text-embeddings

**Description**

`step_pretrained_text_embedding` creates a *specification* of a recipe step that will transform text data into its numerical transformation based on a pretrained model.

**Usage**

`step_pretrained_text_embedding(
  recipe,
  ...,  
  role = "predictor",
  trained = FALSE,
  handle,
  args = NULL,
  skip = FALSE,
  id = recipes::rand_id("pretrained_text_embedding")
)`
Arguments

recipe  A recipe object. The step will be added to the sequence of operations for this recipe.

...  One or more selector functions to choose variables.

role  Role for the created variables

trained  A logical to indicate if the quantities for preprocessing have been estimated.

handle  the Module handle to resolve.

args  other arguments passed to [hub_load()].

skip  A logical. Should the step be skipped when the recipe is baked by [recipes::bake.recipe()]

While all operations are baked when [recipes::prep.recipe()] is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using 'skip = TRUE' as it may affect the computations for subsequent operations

id  A character string that is unique to this step to identify it.

Examples

```r
## Not run:
library(tibble)
library(recipes)
df <- tibble(text = c('hi', "heello", "goodbye"), y = 0)

rec <- recipe(y ~ text, df)
rec <- rec %>% step_pretrained_text_embedding(
  text,
  handle = "https://tfhub.dev/google/tf2-preview/gnews-swivel-20dim-with-oov/1"
)

## End(Not run)
```
Index

bake.step_pretrained_text_embedding, 2
hub_image_embedding_column, 2
hub_load, 3
hub_sparse_text_embedding_column, 4
hub_text_embedding_column, 5
install_tfhub, 5
layer_hub, 6
prep.step_pretrained_text_embedding, 7
step_pretrained_text_embedding, 7