Module: tf.contrib

Contents

Modules

Defined in `tensorflow/contrib/__init__.py`.

contrib module containing volatile or experimental code.

Modules

`bayesflow` module: Ops for representing Bayesian computation.

`cloud` module: Module for cloud ops.

`cluster_resolver` module

`compiler` module: A module for controlling the Tensorflow/XLA JIT compiler.

`copy_graph` module: Functions to copy elements between graphs.

`crf` module: Linear-chain CRF layer.

`cudnn_rnn` module: Ops for fused Cudnn RNN models.

`data` module: `tf.contrib.data.Dataset` API for input pipelines.

`decision_trees` module: Shared representations for tree-based models in tensorflow.

`deprecated` module: Non-core alias for the deprecated tf.X_summary ops.

`distributions` module: Classes representing statistical distributions and ops for working with them.

`estimator` module: Experimental utilities re:tf.estimator.*.

`factorization` module: Ops and modules related to factorization.

`ffmpeg` module: Working with audio using FFmpeg.

`framework` module: Framework utilities.

`gan` module: TFGAN grouped API. Please see README.md for details and usage.

`graph_editor` module: TensorFlow Graph Editor.

`grid_rnn` module: GridRNN cells

`image` module: Ops for image manipulation.

`input_pipeline` module: Ops and modules related to input_pipeline.

`integrate` module: Integration and ODE solvers.

`keras` module: Implementation of the Keras API meant to be a high-level API for TensorFlow.

`kernel_methods` module: Ops and estimators that enable explicit kernel methods in TensorFlow.

`kfac` module: Kronecker-factored Approximate Curvature Optimizer.

`labeled_tensor` module: Labels for TensorFlow.

`layers` module: Ops for building neural network layers, regularizers, summaries, etc.

`learn` module: High level API for learning.

`legacy_seq2seq` module: Deprecated library for creating sequence-to-sequence models in TensorFlow.

`linalg` module: Linear algebra libraries.

`linear_optimizer` module: Ops for training linear models.

`lookup` module: Ops for lookup operations.

`losses` module: Ops for building neural network losses.

`memory_stats` module: Ops for memory statistics.

`meta_graph_transform` module: Utility for applying the Graph Transform tool to a MetaGraphDef.

`metrics` module: Ops for evaluation metrics and summary statistics.

`nccl` module: Functions for using NVIDIA nccl collective ops.

`ndlstm` module: Init file, giving convenient access to all ndlstm ops.

`nn` module: Module for variants of ops in tf.nn.

`opt` module: A module containing optimization routines.

`predictor` module: Modules for `Predictor`s.

`quantization` module: Ops for building quantized models.

`quantize` module: Functions for rewriting graphs for quantized training.

`reduce_slice_ops` module: reduce by slice

`remote_fused_graph` module: Remote fused graph ops python library.

`resampler` module: Ops and modules related to resampler.

`rnn` module: RNN Cells and additional RNN operations.

`saved_model` module: SavedModel contrib support.

`seq2seq` module: Ops for building neural network seq2seq decoders and losses.

`session_bundle` module

`signal` module: Signal processing operations.

`slim` module: Slim is an interface to contrib functions, examples and models.

`solvers` module: Ops for representing Bayesian computation.

`sparsemax` module: Module that implements sparsemax and sparsemax loss, see [1].

`specs` module: Init file, giving convenient access to all specs ops.

`staging` module: contrib module containing StagingArea.

`stat_summarizer` module: Exposes the Python wrapper for StatSummarizer utility class.

`stateless` module: Stateless random ops which take seed as a tensor input.

`tensor_forest` module: Random forest implementation in tensorflow.

`tensorboard` module: tensorboard module containing volatile or experimental code.

`testing` module: Testing utilities.

`tfprof` module: tfprof is a tool that profile various aspect of TensorFlow model.

`timeseries` module: A time series library in TensorFlow (TFTS).

`tpu` module: Ops related to Tensor Processing Units.

`training` module: Training and input utilities.

`util` module: Utilities for dealing with Tensors.

Last updated November 2, 2017.