

## Models in TensorFlow from GitHub

# 图像处理/识别

### 1.PixelCNN & PixelRNN in TensorFlow

TensorFlow implementation of [Pixel Recurrent Neural Networks](#).

地址: <https://github.com/carpedm20/pixel-rnn-tensorflow>

### 2.Simulated+Unsupervised (S+U) learning in TensorFlow

TensorFlow implementation of [Learning from Simulated and Unsupervised Images through Adversarial Training](#).

地址: <https://github.com/carpedm20/simulated-unsupervised-tensorflow>

### 3.ResNet in TensorFlow

Implementation of [Deep Residual Learning for Image Recognition](#). Includes a tool to use He et al's published trained Caffe weights in TensorFlow.

地址: <https://github.com/ry/tensorflow-resnet>

### 4.A composable Generative Adversarial Network(GAN) with API and command line tool

HyperGAN, A versatile GAN(generative adversarial network) implementation focused on scalability and ease-of-use.

地址: <https://github.com/255BITS/HyperGAN>

### 5.conversation of caffe vgg16 model to tensorflow

VGG-16 is my favorite image classification model to run because of its simplicity and accuracy. The creators of this model published [a pre-trained binary](#) that can be used in Caffe.

地址: <https://github.com/ry/tensorflow-vgg16>

### 6.A Kitti Road Segmentation model implemented in tensorflow

KittiSeg performs segmentation of roads by utilizing an FCN based model.

The model achieved [first place](#) on the Kitti Road Detection Benchmark at submission time. Check out our [paper](#) for a detailed model description.

地址: <https://github.com/MarvinTeichmann/KittiSeg>

### 7.TensorFlow tutorial on Generative Adversarial Models

地址: [https://github.com/ericjang/genadv\\_tutorial](https://github.com/ericjang/genadv_tutorial)

### 8.Pretrained models for TFLearn and TensorFlow

地址: <https://github.com/tflearn/models>

### 9. Generative Models with TensorFlow

地址: <https://github.com/arahuja/generative-tf>

### 10. Re-implementation of the m-RNN model using TensorFlow

This package is a re-implementation of the [m-RNN](#) image captioning method using [TensorFlow](#). The training speed is optimized with buckets of different lengths of the training sentences. It also support the *Beam Search* method to decode image features into sentences.

地址: <https://github.com/mjhucla/TF-mRNN>

### 11. Recurrent Models of Visual Attention

Modified from <https://github.com/jlindsey15/RAM>

Implementation of "Recurrent Models of Visual Attention" V. Mnih et al.

Run by python ram.py and it can reproduce the result on Table 1 (a) 28x28 MNIST

地址: <https://github.com/zhongwen/RAM>

### 12. Simple Image Classification Models for the CIFAR-10 dataset using TensorFlow

This is the code for the blog post '[How to Build a Simple Image Recognition System Using TensorFlow](#)'.

地址: <https://github.com/wolfib/image-classification-CIFAR10-tf>

### 13. IllustrationGAN

A simple, clean TensorFlow implementation of Generative Adversarial Networks with a focus on modeling illustrations.

地址: <https://github.com/tdrussell/IllustrationGAN>

### 14. ImageNet pre-trained models with batch normalization

This repository contains convolutional neural network (CNN) models trained on ImageNet by Marcel Simon at the Computer Vision Group Jena (CVGJ) using the Caffe framework. Each model is in a separate subfolder and contains everything needed to reproduce the results. This repository focuses currently contains the batch-normalization-variants of AlexNet and VGG19 as well as the training code for Residual Networks (Resnet).

地址: <https://github.com/cvjena/cnn-models>

### 15. Face recognition using Tensorflow

This is a TensorFlow implementation of the face recognizer described in the paper "[FaceNet: A Unified Embedding for Face Recognition and Clustering](#)".

The project also uses ideas from the paper "[A Discriminative Feature Learning Approach for Deep Face Recognition](#)" as well as the paper "[Deep Face Recognition](#)" from the [Visual Geometry Group](#) at Oxford.

地址: <https://github.com/davidsandberg/facenet>

# 语音/语义/文字

## 1.Multi-layer Recurrent Neural Networks (LSTM, RNN) for word-level language models in Python using TensorFlow

Mostly reused code from <https://github.com/sherjilozair/char-rnn-tensorflow> which was inspired from Andrej Karpathy's [char-rnn](#).

地址: <https://github.com/hunkim/word-rnn-tensorflow>

## 2.LSTM language model with CNN over characters in TensorFlow

Tensorflow implementation of [Character-Aware Neural Language Models](#).

The original code of author can be found [here](#).

地址: <https://github.com/carpedm20/lstm-char-cnn-tensorflow>

## 3.A neural conversational model

My tensorflow implementation of "A neural conversational model", a Deep learning based chatbot. This work tries to reproduce the results of [A Neural Conversational Model](#) (aka the Google chatbot). It uses a RNN (seq2seq model) for sentence predictions. It is done using python and TensorFlow.

地址: <https://github.com/Conchylicultor/DeepQA>

## 4.Tensorflow based Neural Conversation Models

This implementation contains an extension of seq2seq tutorial for conversation models in Tensorflow. Examples of basic model can be found in [this paper](#).

地址: [https://github.com/pbhatia243/Neural\\_Conversation\\_Models](https://github.com/pbhatia243/Neural_Conversation_Models)

## 5.ByteNet for character-level language modelling

This is a tensorflow implementation of the byte-net model from DeepMind's paper [Neural Machine Translation in Linear Time](#).

地址: <https://github.com/paarthneekhara/byteNet-tensorflow>

## 6.Language Modeling with Gated Convolutional Networks

This is a Tensorflow implementation of Facebook AI Research Lab's paper: [Language Modeling with Gated Convolutional Networks](#). This paper applies a convolutional approach to language modelling with a novel Gated-CNN model.

地址: <https://github.com/anantzoid/Language-Modeling-GatedCNN>

## 7.Experiment diverse Deep learning models for music generation with TensorFlow

The different models and experiments are explained [here](#).

地址: <https://github.com/Conchylicultor/MusicGenerator>

## 8.TensorFlow RNN Language Model

This module is an example of how create a recursive neural network language model using [TensorFlow](#).

地址: <https://github.com/wpm/tfrnnlm>

## 9.tensorflow port of the lda2vec model for unsupervised learning of document + topic + word embeddings

TensorFlow implementation of Christopher Moody's [lda2vec](#), a hybrid of [Latent Dirichlet Allocation](#) & [word2vec](#).

地址: <https://github.com/meereem/lda2vec-tf>

## 10.Implement character-level language models for text generation based-on LSTM, in Python/TensorFlow

本程序用于自动生成一段中文文本（训练语料是英文时也可用于生成英文文本），具体生成文本的内容和形式取决于训练语料。模型基本思想和 [karpathy](#) 的 [char-rnn](#) 程序一致，利用循环神经网络 (RNN) 在大规模语料上训练一个 language model，然后利用训练好的 language model 去自动生成一段文本。相比于 theano 版本的 [char-rnn](#) 模型，本模型采用了多层 RNN 而不是单层（tensorflow 中实现一个多层 RNN 简直太方便了），同时还支持 max、sample 和 beam-search 多种生成策略。本程序代码参考了 tensorflow 官方给出的一个 language model 程序 [ptb\\_word\\_lm.py](#)。

地址: <https://github.com/hit-computer/char-rnn-tf>

## 11.Visual Question Answering Demo on pretrained model

This is a simple Demo of Visual Question answering which uses pretrained models (see models/CNN and models/VQA) to answer a given question about the given image.

地址: [https://github.com/iamaaditya/VQA\\_Demo](https://github.com/iamaaditya/VQA_Demo)

## 12.tf-adaptive-softmax-lstm-lm

This repository shows the experiment result of LSTM language models on PTB (Penn Treebank) and GBW ([Google One Billion Word](#)) using AdaptiveSoftmax on TensorFlow.

地址: <https://github.com/TencentAILab/tf-adaptive-softmax-lstm-lm>

# 综合

## 1.TensorFlow Models

This repository contains machine learning models implemented in [TensorFlow](#). The models are maintained by their respective authors.

地址: <https://github.com/tensorflow/models>

## 2.Collection of generative models, e.g. GAN, VAE in Pytorch and Tensorflow

What's in it?

**GAN:** 1.Vanilla GAN 2.Conditional GAN 3.InfoGAN 4.Wasserstein GAN 5.Mode Regularized GAN 6.Coupled GAN 7.Auxiliary Classifier GAN 8.Least Squares GAN 9.Boundary Seeking GAN 10.Energy Based GAN 11.f-GAN 12.Generative Adversarial Parallelization 12.DiscoGAN 13Adversarial Feature Learning & Adversarially Learned Inference

**VAE:** 1.Vanilla VAE 2.Conditional VAE 3.Denoising VAE 4.Adversarial Autoencoder 5.Adversarial Variational Bayes

地址: <https://github.com/wiseodd/generative-models>

## 3.Deep learning using tensorflow

Tensorflow Projects A repo of everything deep and neurally related.

Implementations and ideas are largely based on papers from arxiv and implementations, tutorials from the internet.

地址: <https://github.com/shekkizh/TensorflowProjects>

## 4.A library for probabilistic modeling, inference, and criticism. Deep generative models, variational inference. Runs on TensorFlow.

[Edward](#) is a Python library for probabilistic modeling, inference, and criticism. It is a testbed for fast experimentation and research with probabilistic models, ranging from classical hierarchical models on small data sets to complex deep probabilistic models on large data sets. Edward fuses three fields: Bayesian statistics and machine learning, deep learning, and probabilistic programming.

地址: <https://github.com/blei-lab/edward>

## 5.Tensorflow Tutorial files and Implementations of various Deep NLP and CV Models.

This repository contains Tensorflow implementations of various deep learning models, with a focus on problems in Natural Language Processing. Each individual subdirectory is self-contained, addressing one specific model.

地址: <https://github.com/siddk/deep-nlp>

## 6.A tensorflow library for building all kinds of models

TensorGraph is a framework for building any imaginable models based on TensorFlow.

As deep learning becomes more and more common and the architectures becoming more and more complicated, it seems that we need some easy to use framework to quickly build these models and that's why TensorGraph is born. It's a very simple and easy to use framework, but it allows you to build all kinds of imaginable models.

地址: <https://github.com/hycis/TensorGraph>

## 7. PyTorch and Tensorflow functional model definitions

Model definitions and pretrained weights for PyTorch and Tensorflow. PyTorch, unlike lua torch, has autograd in its core, so using modular structure of torch.nn modules is not necessary, one can easily allocate needed variables and write a function that utilizes them, which is sometimes more convenient. This repo contains model definitions in this functional way, with pretrained weights for some models.

Weights are serialized as a dict of arrays in hdf5, so should be easily loadable in other frameworks. Thanks to @edgarriba we have [cpp\\_parser](#) for loading weights in C++.

More models coming! We also plan to add definitions for other frameworks in future, probably tiny-dnn first. Contributions are welcome.

See also imagenet classification with PyTorch [demo.ipynb](#)

地址: <https://github.com/szagoruyko/functional-zoo>

## 8. Neural network models in tensorflow

地址: <https://github.com/AJwader/Tensorflow-models>

# 其他

## 1. Caffe models in TensorFlow

Convert [Caffe](#) models to [TensorFlow](#).

地址: <https://github.com/ethereon/caffe-tensorflow>

## 2. Run Keras models (tensorflow backend) in the browser, with GPU support

Models are created directly from the Keras JSON-format configuration file, using weights serialized directly from the corresponding HDF5 file. Also works in node, but only in CPU mode.

地址: <https://github.com/transcranial/keras-js>

## 3. Simplify the training and tuning of Tensorflow models

Stop wasting your time rewriting the training, evaluation & visualization procedures for your ML model: let DyTB do the work for you!

地址: <https://github.com/galeone/dynamic-training-bench>

## 4. Observations and notes to understand the workings of neural network models and other thought experiments using Tensorflow

A repo of observations and notes to understand the workings of neural network models and other simple thought experiments using Tensorflow.

地址: <https://github.com/shekkizh/neuralnetworks.thought-experiments>

## 5. attention model for entailment on SNLI corpus implemented in Tensorflow and Keras

Implementations of a attention model for entailment from [this paper](#) in keras and tensorflow.

Compatible with keras v1.0.6 and tensorflow 0.11.0rc2

I implemented the model to learn the APIs for keras and tensorflow, so I have not really tuned on the performance. The models implemented in keras is a little different, as keras does not expose a method to set a LSTMs state.

地址: <https://github.com/shyamupa/snli-entailment>

## **6.Multilayer Feed-Forward Neural Network predictive model implementations with TensorFlow and scikit-learn**

This project provides multilayer perceptron predictive models, implemented using [TensorFlow](#) and following the [scikit-learnPredictor API](#).

地址: <https://github.com/civisanalytics/muffnn>

## **7.Keras pretrained models (VGG16 and InceptionV3) + Transfer Learning for predicting classes in the Oxford 102 flower dataset**

See my application for identifying plants and taking care them - [Plant Care](#). It works using the code from the model implemented in this repo.

This bootstraps the training of deep convolutional neural networks with [Keras](#) to classify images in the [Oxford 102 category flower dataset](#).

Train process is fully automated and the best weights for the model will be saved.

This code can be used for any dataset, just follow the original files structure in data/sorted directory after running `bootstrap.py`. If you wish to store your dataset somewhere else, you can do it and run `train.py` with setting a path to dataset with a special parameter `--data_dir==path/to/your/sorted/data`.

地址: <https://github.com/Arsey/keras-transfer-learning-for-oxford102>

## **8.Tensorflow Model Zoo for Torch7 and PyTorch**

This is a porting of tensorflow pretrained models made by [Remi Cadene](#) and [Micael Carvalho](#). Special thanks to Moustapha Cissé. All models have been tested on Imagenet.

This work was inspired by [inception-v3.torch](#).

地址: <https://github.com/Cadene/tensorflow-model-zoo.torch>

## **9.Keras implementation of "Wide Residual Networks"**

This repo contains the code to run Wide Residual Networks using Keras.

- Paper (v1): <http://arxiv.org/abs/1605.07146v1> (the authors have since published a v2 of the paper, which introduces slightly different preprocessing and improves the accuracy a little).
- Original code: <https://github.com/szagoruyko/wide-residual-networks>

地址: [https://github.com/asmith26/wide\\_resnets\\_keras](https://github.com/asmith26/wide_resnets_keras)