



TensorFlow Official Models

The TensorFlow official models are a collection of models that use TensorFlow's high-level APIs. They are intended to be well-maintained, tested, and kept up to date with the latest TensorFlow API.

They should also be reasonably optimized for fast performance while still being easy to read. These models are used as end-to-end tests, ensuring that the models run with the same or improved speed and performance with each new TensorFlow build.

More models to come!

The team is actively developing new models. In the near future, we will add:

- State-of-the-art language understanding models.
- State-of-the-art image classification models.
- State-of-the-art object detection and instance segmentation models.
- State-of-the-art video classification models.

Table of Contents

- [Models and Implementations](#)
 - [Computer Vision](#)
 - [Image Classification](#)
 - [Object Detection and Segmentation](#)
 - [Video Classification](#)
 - [Natural Language Processing](#)
 - [Recommendation](#)
- [How to get started with the official models](#)
- [Contributions](#)

Models and Implementations

Computer Vision

Image Classification

Model	Reference (Paper)
MNIST	A basic model to classify digits from the MNIST dataset
ResNet	Deep Residual Learning for Image Recognition
ResNet-RS	Revisiting ResNets: Improved Training and Scaling Strategies
EfficientNet	EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks
Vision Transformer	An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

Object Detection and Segmentation

Model	Reference (Paper)
RetinaNet	Focal Loss for Dense Object Detection
Mask R-CNN	Mask R-CNN
ShapeMask	ShapeMask: Learning to Segment Novel Objects by Refining Shape Priors
SpineNet	SpineNet: Learning Scale-Permuted Backbone for Recognition and Localization
Cascade RCNN-RS and RetinaNet-RS	Simple Training Strategies and Model Scaling for Object Detection

Video Classification

Model	Reference (Paper)
Mobile Video Networks (MoViNets)	MoViNets: Mobile Video Networks for Efficient Video Recognition

Natural Language Processing

Model	Reference (Paper)
ALBERT (A Lite BERT)	ALBERT: A Lite BERT for Self-supervised Learning of Language Representations
BERT (Bidirectional Encoder Representations from Transformers)	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
NHNet (News Headline generation model)	Generating Representative Headlines for News Stories
Transformer	Attention Is All You Need
XLNet	XLNet: Generalized Autoregressive Pretraining for Language Understanding
MobileBERT	MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices

Recommendation

Model	Reference (Paper)
DLRM	Deep Learning Recommendation Model for Personalization and Recommendation Systems
DCN v2	Improved Deep & Cross Network and Practical Lessons for Web-scale Learning to Rank Systems
NCF	Neural Collaborative Filtering

How to get started with the official models

- The official models in the master branch are developed using [master branch of TensorFlow 2](#). When you clone (the repository) or download (`pip` binary) master branch of official models , master branch of

TensorFlow gets downloaded as a dependency. This is equivalent to the following.

```
pip3 install tf-models-nightly
pip3 install tensorflow-text-nightly # when model uses `nlp` packages
```

- In case of stable versions, targeting a specific release, Tensorflow-models repository version numbers match with the target TensorFlow release. For example, [TensorFlow-models v2.5.0] (<https://github.com/tensorflow/models/releases/tag/v2.5.0>) is compatible with [TensorFlow v2.5.0] (<https://github.com/tensorflow/tensorflow/releases/tag/v2.5.0>). This is equivalent to the following.

```
pip3 install tf-models-official==2.5.0
pip3 install tensorflow-text==2.5.0 # when model uses `nlp` packages
```

Please follow the below steps before running models in this repository.

Requirements

- The latest TensorFlow Model Garden release and the latest TensorFlow 2
 - If you are on a version of TensorFlow earlier than 2.2, please upgrade your TensorFlow to [the latest TensorFlow 2](#).
- Python 3.7+

Our integration tests run with Python 3.7. Although Python 3.6 should work, we don't recommend earlier versions.

Installation

Please check [here](#) for the instructions

Contributions

If you want to contribute, please review the [contribution guidelines](#).