Object Detection API with TensorFlow 1

Requirements

`Python` `3.6` `🍃 TensorFlow` `1.15` `ProtoBuf Compiler` `>3.0`

Installation

You can install the TensorFlow Object Detection API either with Python Package Installer (pip) or Docker. For local runs we recommend using Docker and for Google Cloud runs we recommend using pip.

Clone the TensorFlow Models repository and proceed to one of the installation options.

```
git clone https://github.com/tensorflow/models.git
```

Docker Installation

```
# From the root of the git repository
docker build -f research/object_detection/dockerfiles/tf1/Dockerfile -t od .
docker run -it od
```

Python Package Installation

```
cd models/research
# Compile protos.
protoc object_detection/protos/*.proto --python_out=.
# Install TensorFlow Object Detection API.
cp object_detection/packages/tf1/setup.py .
python -m pip install --use-feature=2020-resolver .
```

```
# Test the installation.
python object_detection/builders/model_builder_tf1_test.py
```

Quick Start

Colabs

- [Jupyter notebook for off-the-shelf inference](#)
- [Training a pet detector](#)

Training and Evaluation

To train and evaluate your models either locally or on Google Cloud see [instructions](#).

Model Zoo

We provide a large collection of models that are trained on several datasets in the [Model Zoo](#).

Guides

- [Configuring an object detection pipeline](#)
- [Preparing inputs](#)
- [Defining your own model architecture](#)
- [Bringing in your own dataset](#)
- [Supported object detection evaluation protocols](#)
- [TPU compatible detection pipelines](#)
- [Training and evaluation guide (CPU, GPU, or TPU)](#)

Extras:

- [Exporting a trained model for inference](#)
- [Exporting a trained model for TPU inference](#)
- [Inference and evaluation on the Open Images dataset](#)
- [Run an instance segmentation model](#)
- [Run the evaluation for the Open Images Challenge 2018/2019](#)
- [Running object detection on mobile devices with TensorFlow Lite](#)
- [Context R-CNN documentation for data preparation, training, and export](#)