

Installation

martes, 29 de enero de 2019 15:56

Última Fecha Actualización: 15/03/2019

Objetivo: Pasos para instalación de **TF Object Detection API** en PC con *Windows 10 + Anaconda*.

Fuentes:

- TensorFlow Object Detection API tutorial: <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/>

Pasos de instalación:

1. Configurar ambiente y librerías requeridas:

- Crear environment en Anaconda y activarlo (Para este tutorial se crea environment: **tensorflow**).
- Instalar las siguientes librerías:
 - tensorflow - 1.13.1.
 - pillow - 5.4.1.
 - lxml - 4.3.2.
 - jupyter.
 - matplotlib - 3.0.3.
 - opencv - v3.4.5.20.

Nota: Las librerías anteriores se instalan todas por medio del comando: **pip install <librería>**.

Para opencv el comando es: **pip install opencv-python (==3.4.5.20)**.

2. Descargar TensorFlow Models:

- Crear una nueva carpeta llamada *tensorflow* en la ubicación de preferencia (Para este tutorial se crea en *C:\tensorflow*) y navegar a esa carpeta en Anaconda Prompt.
- Clonar proyecto de [TensorFlow Models](#) en el directorio local como se muestra a continuación:

```
Anaconda Prompt - git clone https://github.com/tensorflow/models.git

(tensorflow) C:\tensorflow>git clone https://github.com/tensorflow/models.git
Cloning into 'models'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (13/13), done.
Receiving objects: 19% (4769/24603), 20.16 MiB | 1.28 MiB/s
```

- Una vez clonado el proyecto, dentro de la carpeta *models* se debe tener la siguiente estructura:

| OS (C:) > tensorflow > models | | | |
|-------------------------------|-----------|------------------|-------------|
| | Name | Date modified | Type |
| | .git | 08/03/2019 16:12 | File folder |
| | official | 08/03/2019 16:12 | File folder |
| | research | 08/03/2019 16:12 | File folder |
| | samples | 08/03/2019 16:12 | File folder |
| | tutorials | 08/03/2019 16:12 | File folder |

3. Instalación Protobuf:

TensorFlow Object Detection API usa Protobufs para configurar los parámetros del modelo y entrenamiento. Antes de usar el API de Object Detection, las librerías de Protobuf deben ser descargadas y compiladas.

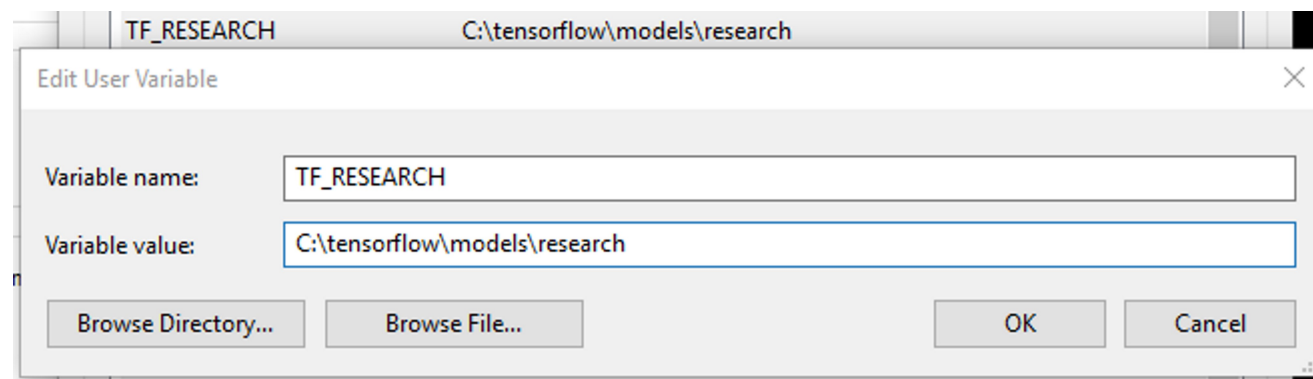
- Descargar la última versión de "[Protocol Buffers](#)" (Para este tutorial se descarga la versión "protoc-3.7.0-win64.zip").
- Crear una carpeta que se llame "Google Protobuf" en la ruta "C:\Program Files\Google Protobuf" y descomprimir el archivo descargado en el punto anterior en esta ubicación.
- Adicionar "C:\Program Files\Google Protobuf\bin" a la variable de entorno *Path*.
- Reiniciar la consola Anaconda Prompt (Por cambios en las variables de entorno) y navegar a la ruta "C:\tensorflow\models\research\".
- Ejecutar el siguiente comando **"for /f %i in ('dir /b object_detection\protos*.proto') do protoc object_detection\protos\%i --python_out=."**:

```
Anaconda Prompt

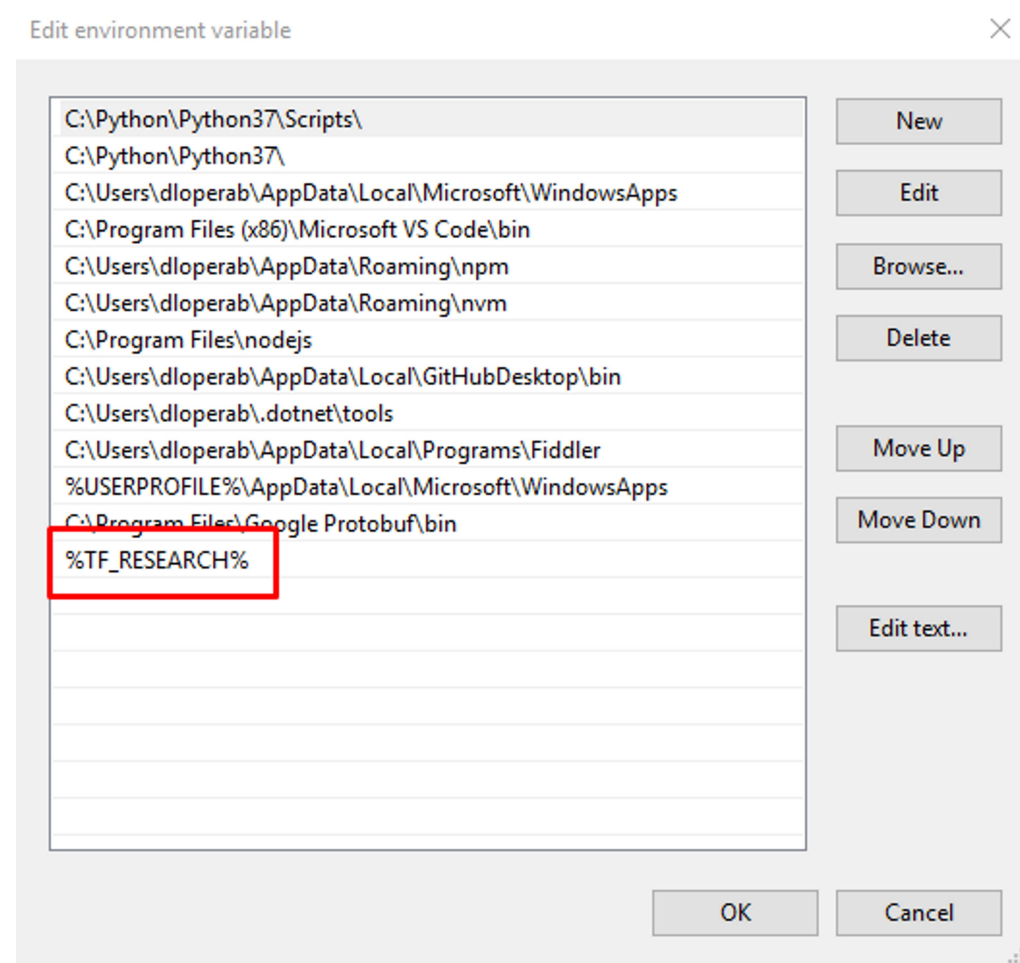
(tensorflow) C:\tensorflow\models\research>for /f %i in ('dir /b object_detection\protos\*.proto') do protoc object_detection\protos\%i --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\anchor_generator.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\argmax_matcher.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\bipartite_matcher.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\box_coder.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\box_predictor.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\calibration.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\eval.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\faster_rcnn.proto --python_out=.
(tensorflow) C:\tensorflow\models\research>protoc object_detection\protos\faster_rcnn_box_coder.proto --python_out=.
```

4. Adicionar variables de entorno necesarias:

- Adicionar el paquete de *research* del API de object detection "<PATH_TO_TF>\models\research\" en una variable de sistema. En el caso de este tutorial se crea la variable del sistema "TF_RESEARCH":



- Adicionalmente, esta variable se adiciona la variable de entorno "Path":



- **TEST Installationn:** <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#test-tf-models>
- **Labelling Installation:** <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/install.html#labelling-installation>