

Tensorflow-Keras

Siham Tabik

Dpto. Ciencias de la Computación e I.A.

Universidad de Granada

siham@ugr.es



**UNIVERSIDAD
DE GRANADA**



Esquema

- Librerías para programar redes neuronales
- ¿Porqué Tensorflow?
- Modelo de programación de tf.keras
- Estructura de tf.keras
- ¿Cómo programar con tf.keras?
- Ejemplos

Librerías para programar Redes Neuronales



theano



PYTORCH

Popularidad Según Google Trends

● Tensorflow + Keras
Término de búsqueda

● PyTorch
Aplicación

● MXNET
Término de búsqueda

● Theano
Término de búsqueda

+

España ▾

Últimos 12 meses ▾

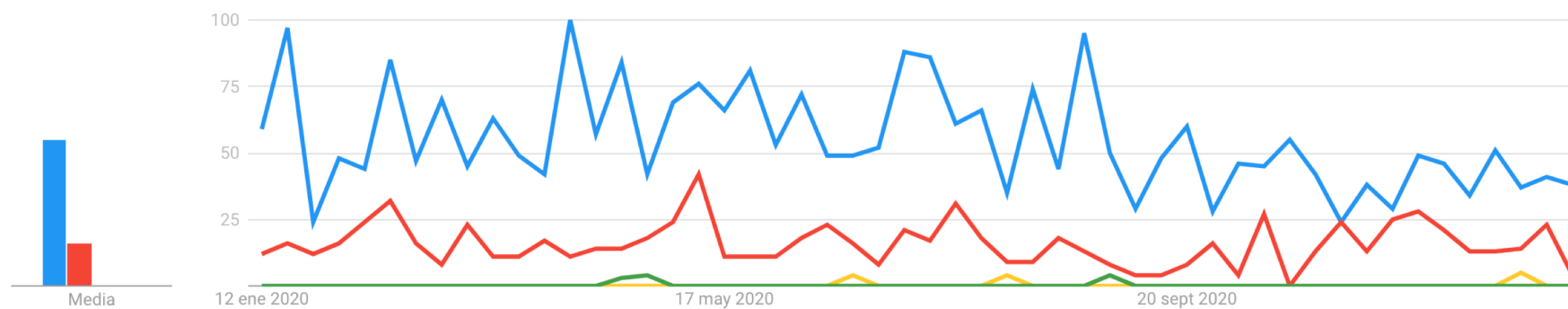
Todas las categorías ▾

Búsqueda web ▾

! **Nota:** Esta comparación contiene temas y términos de búsqueda, los cuales se miden de forma diferente.

[MÁS INFORMACIÓN](#)

Interés a lo largo del tiempo ?

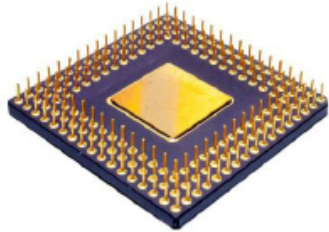


¿Por qué Tensorflow?

- Es un **ecosistema** de herramientas, librerías y recursos para la construcción y despliegue de los modelos de ML
- El usuario/programador se tiene que centrar a seleccionar y ensamblar los módulos/bloques pre-fabricados
- Existe muchos recursos, e.g., tutorials, videos, códigos, entornos (colab) <https://www.tensorflow.org/tutorials/>
- El cálculo del gradiente y backpropagation está oculto al usuario
- Open-source, gran comunidad (+12,000 commits and +5000 TF-related repos since Nov. 2015), +570 colaboradores
- Se usa en cursos de ML en muchas universidades: Berkeley, Stanford, Toronto
- Proyectos impresionantes que ya usan TensorFlow

Tensorflow es Protable

Desplegar los códigos TF a uno o más CPUs o GPUs de un sobremesa, servidor o smartphone con [TensorFlow Lite](#) & [TensorFlow.js](#)



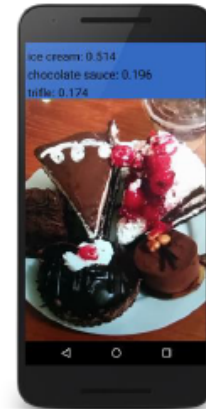
CPU



TPU



Android



iOS



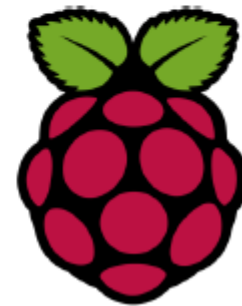
Raspberry
Pi



GPU

Tensorflow Flexible

From Raspberry Pi, Android, Windows, iOS, Linux to server farms



Tensorflow se usa por:



¿Cómo programar con Tensorflow?

El usuario puede usar en dos niveles:

- **Nivel bajo:** `tf.*` , usar grafos computacionales, tensores y *sessions*
- **Nivel alto:** [tf.keras](#)

Redes neuronales profundas con tf.keras

Capa configurable



⋮

Capa configurable



⋮



tf.keras

- Framework de alto nivel para construir redes neuronales
- Escrito en Python
- Integrado en Tensorflow 2.0
- Diseñada para construir redes neuronales de forma simple y con pocas líneas de código
- Conectando varios tipos de bloques/capas configurables
- Es altamente modular y usa Python
- La idea básica: Un modelo es una secuencia layers configurables, que pueden conectarse entre si.

Modelo de programación de tf.keras

La estructura del código es:

1. Cargar los datos
2. Definir el modelo
3. Compilar el modelo
4. Ajustar, in ingles, “*Fit*” el modelo
5. Evaluar el modelo

Modelo de programación de tf.keras

```
(train_images, train_labels), (test_images, test_labels) = mnist.load_data()

model = keras.models.Sequential()
model.add(keras.layers.Flatten(input_shape=(28, 28)))
model.add(keras.layers.Dense(128, activation='relu'))
model.add(keras.layers.Dense(10, activation='softmax'))

model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

model.fit(train_images, train_labels, epochs=10)

test_loss, test_acc = model.evaluate(test_images, test_labels, verbose=2)
```

Models, Applications, Capas, Datasets

Models

- Sequential()
- Model()

Applications

- Modelos pre-entrenados
- De arquitecturas potentes, DenseNet, EfficientNet, inception_resnet_v2

Tipo de capas/layers:

- **Core layers:** Dense, activation, dropout, flatten, input, reshape.
- **Convolutional layers:** Conv2D
- **Pooling layers:** MaxPooling2D
- **Recurrent layers:** LSTM
- **Embedding layers:** Embedding
- **Normalization layers:** BatchNormalization

Datasets

- `tf.data.Dataset: tf.data.Dataset.mnist, cifar`
- <https://www.tensorflow.org/datasets/catalog/overview?hl=es-419>

Modelos, *applications*, capas ...

Modules

[applications](#) : Keras Applications are canned architectures with pre-trained weights.

[callbacks](#) : Callbacks: utilities called at certain points during model training.

[datasets](#) : Public API for tf.keras.datasets namespace.

[initializers](#) : Keras initializer serialization / deserialization.

[layers](#) : Keras layers API.

[losses](#) : Built-in loss functions.

[metrics](#) : Built-in metrics.

[models](#) : Code for model cloning, plus model-related API entries.

[optimizers](#) : Built-in optimizer classes.

[preprocessing](#) : Keras data preprocessing utils.

[regularizers](#) : Built-in regularizers.

[utils](#) : Public API for tf.keras.utils namespace.

...

Classes

[class Model](#): Model groups layers into an object with training and inference features.

[class Sequential](#): Linear stack of layers.

Functions

[Input\(...\)](#): Input() is used to instantiate a Keras tensor.

¿Qué es Colab?

- Una herramienta gratuita que a través del navegador *Chrome* o *FireFox*
- Permite usar y compartir blocs de notas de Jupyter con otras personas sin tener que descargar, instalar ni ejecutar nada en la computadora.
- Jupyter se basa en Colaboratory
- Es equivalente a un googledoc pero para desarrollar códigos

Link: <https://research.google.com/colaboratory/faq.html>