

Tema 3.1

Software para Deep Learning

Miguel Ángel Martínez del Amor

Deep Learning

Departamento Ciencias de la Computación e Inteligencia Artificial

Universidad de Sevilla

Contenido

- Niveles de programación
- Ecosistema actual

Niveles de programación

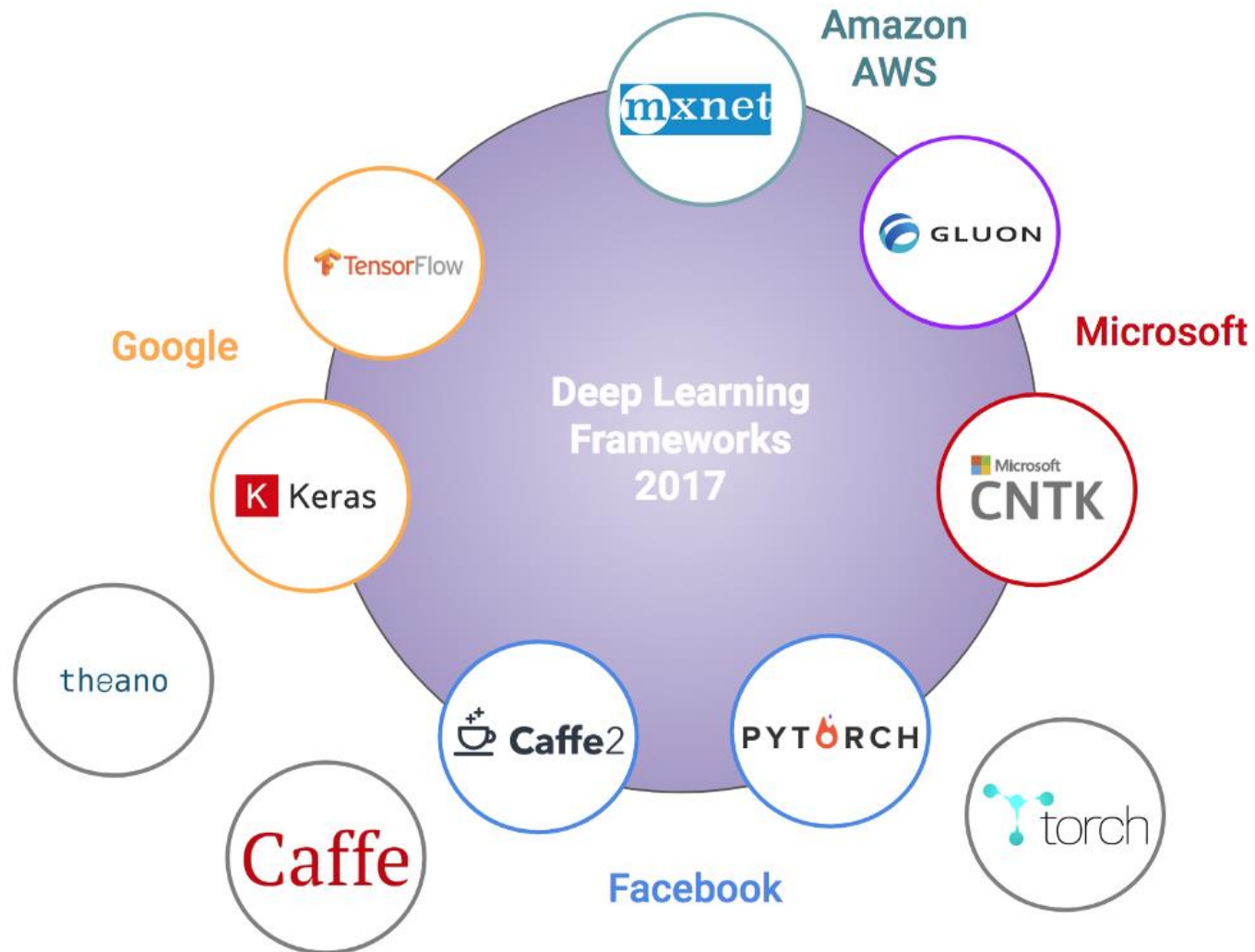
- La mejor forma de trabajar con redes neuronales es simular los modelos en un ordenador.
- Programación a nivel 0:
 - Podemos elegir el lenguaje de nuestra elección, una buena tarde e implementar los conceptos.
 - ¿Qué pasa si quiero cambiar la arquitectura de la red? Si no lo he hecho bien, tendría que re-programarla desde cero, sobre todo para ajustar la propagación del gradiente.

Niveles de programación

- Programación a nivel 1:
 - APIs que proveen de bloques que podemos usar.
 - Tenemos que encajar las piezas para construir nuestro modelo.
 - Librerías de diferenciación automática.
- Programación a nivel 2:
 - APIs que proveen una capa de abstracción sobre los detalles de modelo.
 - Encajamos bloques que representan partes de la arquitectura que queremos montar.



Ecosistema actual



Ecosistema actual

- **TensorFlow:**

- Creado por **Google** en 2015.
- Usado por grandes compañías: Twitter, Uber, Airbnb, NVIDIA
- Lenguaje principal: **Python**. Soporte también para C++.
- En clusters de ordenadores y en plataformas móviles (iOS, Android).
- Muy flexible: trabaja con un grafo computacional estático.
- Programación a nivel 1: curva de aprendizaje alta.



Ecosistema actual

- **PyTorch:**

- Creado por **Facebook** en 2017. Un porte de Torch a Python.
- Lenguaje principal: **Python**.
- Permite usar depuradores de código Python estándar.
- Trabaja con un grafo computacional dinámico.
- Programación a nivel 1.
- PyTorch compite con TensorFlow directamente.

P Y T  R C H

Ecosistema actual

- **Caffe2:**

- Creado originalmente por Yangqing Jia en la UC Berkeley en 2013.
- La segunda versión fue desarrollada por Facebook y NVIDIA en 2017.
- Programación a nivel 2: prototipado rápido de redes.
- En 2018, Facebook integró Caffe2 en PyTorch.
- API para **C++, Python, MATLAB, y CL**.
- Una gran colección de modelos en [Caffe Model Zoo](#).



Caffe2

Ecosistema actual

- **Keras:**

- Creado por F. Chollet en 2015.
- Programación a nivel 2: prototipado rápido de redes en **Python**.
- Requiere de un backend: Tensorflow, Theano, CNTK.
- Es el mejor framework para comenzar con conocimientos básicos.
- Potente para resolver problemas complejos (muchas soluciones en Kaggle).
- API muy madurada, totalmente integrada ahora en Tensorflow.



Ecosistema actual

- Orientación:
 - Usar **Keras** para arrancar y para prototipado.
 - Para investigación, usar **PyTorch** o **TensorFlow**.
 - Para producción en la nube, usar **TensorFlow** (Google Cloud), **MXNet** (AWS) o **CNTK** (Azure).
 - Para Android, tener en cuenta **DL4J**, y para iOS, **CoreML**.
- [Aquí](#) un resumen de los 10 más usados en 2019
- [Aquí](#) una tabla comparativa de los entornos