



Data Augmentation y Transfer Learning

Deep Learning

Dr. Juan Bekios Calfa

Magíster en Inteligencia Artificial
Facultad de Ingeniería y Ciencias

Información de Contacto

- Juan Bekios Calfa
 - email: juan.bekios@edu.uai.cl, juan.bekios@ucn.cl
 - Web page: <http://jbekios.ucn.cl>
 - Teléfono: 235(5162) - 235(5125)

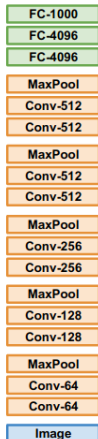
Contenidos

Introducción



Transfer learning en redes convolucionales

1. Entrenar red en Imagenet

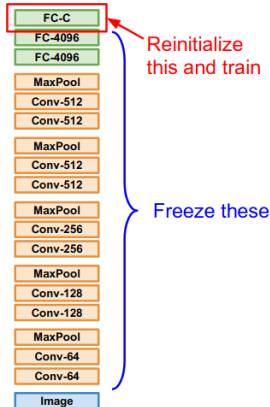


Transfer learning en redes convolucionales

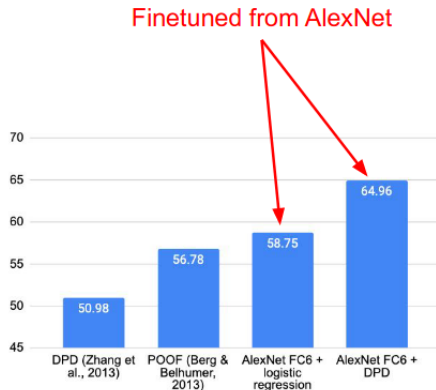
1. Entrenar red con Imagenet



2. Entrenar en dataset pequeño de interés



Transfer learning en redes convolucionales



Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014



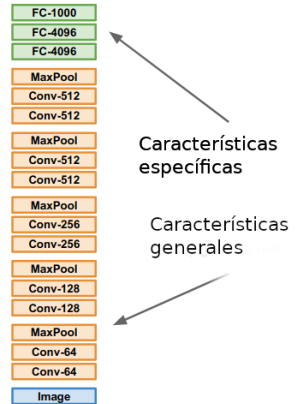
Transfer learning en redes convolucionales

- Con un *set* de datos más grande es posible entrenar más capas de la red convolucional.
- Es necesario utilizar un *ratio* de aprendizaje más pequeño.



Transfer learning en redes convolucionales

	Set de datos similar	Set de datos muy distinto
Pocos datos	?	?
Muchos datos	?	?



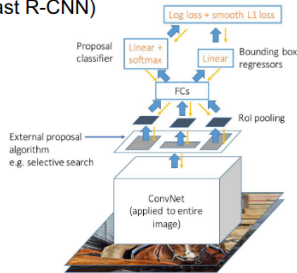
Transfer learning en redes convolucionales

	Set de datos similar	Set de datos muy distinto
Pocos datos	Usa un clasificador lineal en la capa superior	Tienes un problema... intenta utilizar un clasificador no lineal u otras técnicas
Muchos datos	Ajustar unas pocas capas	Ajusta un número más grande de capas

***Transfer learning* con redes convolucionales es común**

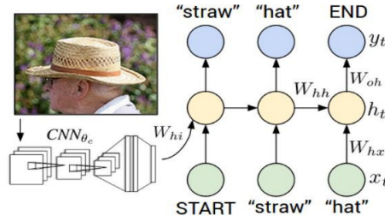
Transfer learning en las redes convolucionales es la regla, no la excepción.

**Object Detection
(Fast R-CNN)**



Girshick, "Fast R-CNN", ICCV 2015
Figure copyright Ross Girshick, 2015. Reproduced with permission.

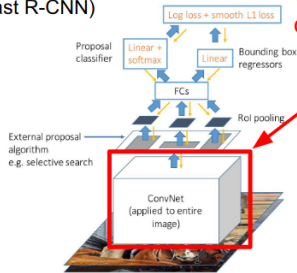
Image Captioning: CNN + RNN



Karpathy and Fei-Fei, "Deep Visual-Semantic Alignments for Generating Image Descriptions", CVPR 2015
Figure copyright IEEE, 2015. Reproduced for educational purposes.

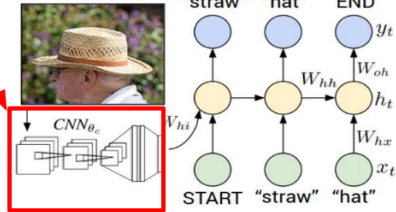
Transfer learning con redes convolucionales es común

Object Detection (Fast R-CNN)



CNN pretrained
on ImageNet

Image Captioning: CNN + RNN



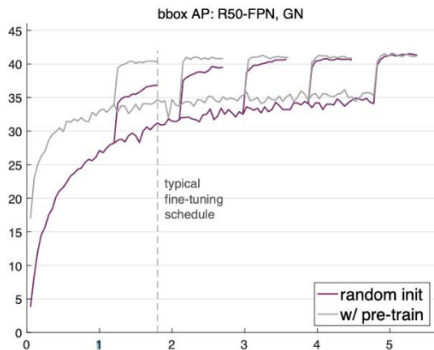
Girshick, "Fast R-CNN", ICCV 2015
Figure copyright Ross Girshick, 2015. Reproduced with permission.

Karpathy and Fei-Fei, "Deep Visual-Semantic Alignments for
Generating Image Descriptions", CVPR 2015
Figure copyright IEEE, 2015. Reproduced for educational purposes.

***Transfer learning* con redes convolucionales es común, pero quizás no siempre es necesario**

El entrenamiento desde cero puede funcionar tan bien como el entrenamiento desde un modelo de ImageNet previamente entrenado para la detección de objetos, pero se tarda entre 2 y 3 veces más en entrenar.

También encuentran que recopilar más datos es mejor que ajustar una red pre entrenada en una tarea relacionada



He et al, "Rethinking ImageNet Pre-training", ICCV 2019
Figure copyright Kaifeng He, 2019. Reproduced with permission.

En resumen

¿Tienes un set de datos de interés con menos de un millón de imágenes?

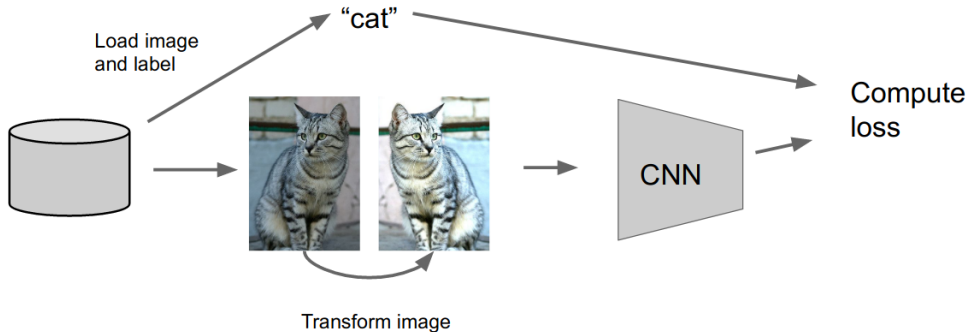
1. Busca un set de datos similar y entrena una red convolucional
2. Haz *transfer learning* y entrena con el set de datos de interés

Los frameworks de *deep learning* proporcionan un "Zoológico de modelos" con modelos previamente entrenados, por lo que no es necesario que se entrenen los suyos.

TensorFlow: <https://github.com/tensorflow/models>

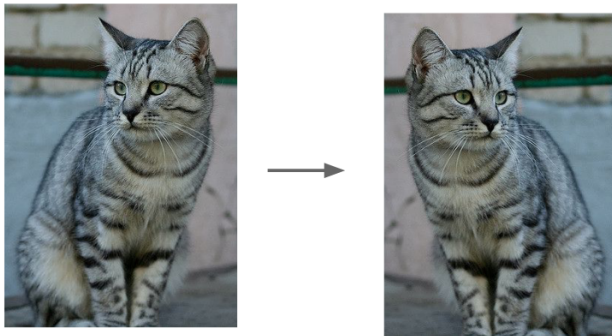
PyTorch: <https://github.com/pytorch/vision>

Data augmentation



Data augmentation

Reflejo horizontal

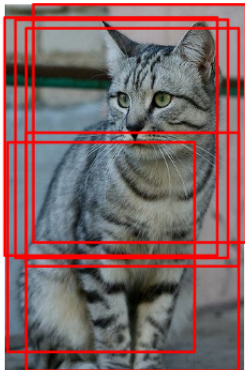


Data augmentation

Cortes y escalas aleatorias

Entrenamiento: Obtener muestras con cortes y escalas aleatorias

1. Seleccionar una imagen L con tamaño $[256, 480]$
2. Cambiar el tamaño de la imagen de entrenamiento, lado corto = L
3. Obtener una muestra de forma aleatoria de una sección de tamaño $[224, 224]$

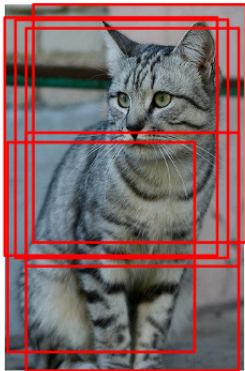


Data augmentation

Cortes y escalas aleatorias

Validación: Promediar un conjunto fijo de cortes

1. Cambiar el tamaño de la imagen a 5 escalas: 224, 256, 384, 480, 640
2. Para cada tamaño, utilice 10 cortes de 224 x 224: 4 esquinas + centro, + reflejo

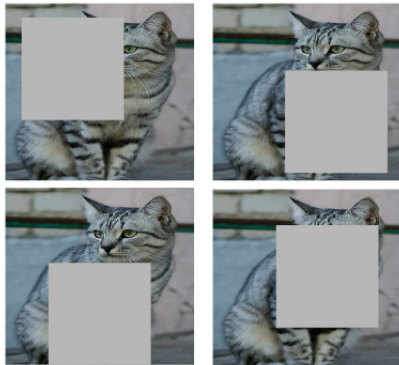


Data augmentation

Cutout

Entrenamiento: Hacer que zonas aleatorias de la imagen sean cero

Validacion: Usar la imagen completa



Data augmentation

Mixup



CNN

Target label:
cat: 0.4
dog: 0.6

Entrenamiento: Utilizar combinaciones aleatorias de dos imagenes

Validacion: Usar la imagen original

Zhang et al, "mixup: Beyond Empirical Risk Minimization", ICLR 2018

¿Preguntas?