

Tema 5.4

Transferencia de Aprendizaje

Deep Learning

Máster Oficial en Ingeniería Informática

Universidad de Sevilla

Contenido

- Transfer Learning (Transferencia de Aprendizaje)
- Estrategias
- Transfer Learning en Deep Learning

Transfer Learning

- Supongamos **nuestra tarea (T1)** es identificar objetos en imágenes de un restaurante (**dominio 1**). Entrenamos un modelo para ello.
- Supongamos **nueva tarea (T2)** es identificar objetos en imágenes de un parque o una cafetería (**dominio 2**). Si aplicamos el modelo entrenado para T1, veremos una degradación de rendimiento para el dominio 2.
- **Transferencia de aprendizaje** debería permitirnos reutilizar el conocimiento de tareas previamente aprendidas y aplicarlas a nuevas.



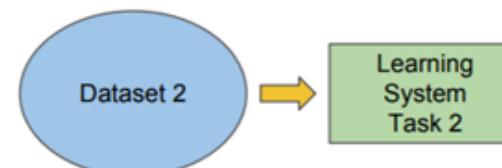
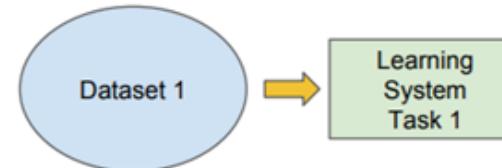
ML tradicional vs Transfer Learning

Traditional ML

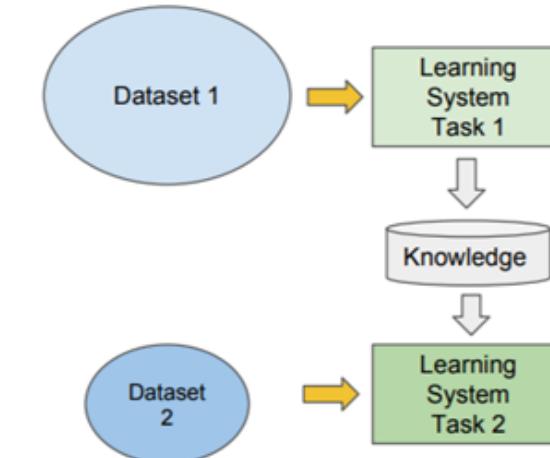
vs

Transfer Learning

- Isolated, single task learning:
 - Knowledge is not retained or accumulated. Learning is performed w.o. considering past learned knowledge in other tasks



- Learning of a new tasks relies on the previous learned tasks:
 - Learning process can be faster, more accurate and/or need less training data



Transfer Learning

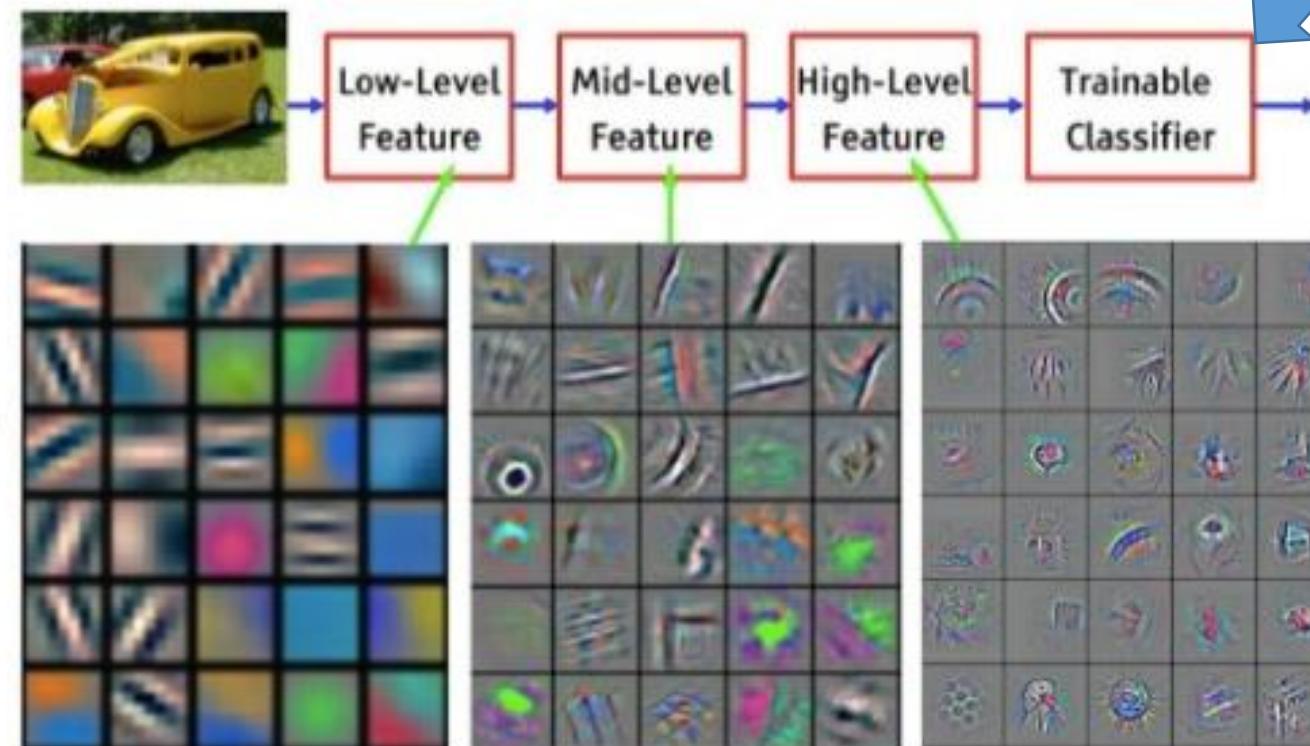
- **Transfer Learning:** habilidad de reutilizar conocimiento existente sobre una tarea origen en otras tareas objetivo.
 - ¿Qué transferir?
 - Identificar qué porción de conocimiento es específico de nuestro origen, y qué tiene en común el origen con el objetivo.
 - ¿Cuándo transferir?
 - Hay algunos escenarios donde transferir conocimiento es contraproducente.
 - ¿Cómo transferir?
 - Buscar los cambios en los algoritmos y las diferentes técnicas existentes para ello.

Transfer Learning en Deep Learning

- Para entrenar una red convolucional (CNN) desde cero se necesita:
 - **Much(ísim)os datos** (*p.ej. ImageNet: 1,2 millones de imágenes, 1000 categorías*)
 - **Gran capacidad computacional** (*p.ej. DGX v2 con 16 Tesla V100*).
 - **Tiempo** (*semanas a meses para entrenamiento*)
- En la realidad, pocos investigadores entranan una CNN desde cero
 - Partir de una ConvNet pre-entrenada en un conjunto de datos muy grande
- Yosinski et al. [How transferable are features in deep neural networks?](#)
2014

Transfer Learning en Deep Learning

Convolutional Neural Network



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Transfer Learning en Deep Learning

- Tres escenarios:
 - **Modelo pre-entrenado (pretrained models)**: descargar un modelo (p.ej. [Model Zoo de Caffe](#)) para usarlo directamente o aplicarle lo anterior lo siguiente.
 - **Extractor de características (fixed feature extractor)**: reemplazar las últimas capas FC (Fully Connected) y el clasificador por nuevas y entrenarlas, fijando el resto.
 - Re-entrenar el clasificador con el nuevo conjunto de datos.
 - **Refinado (fine tuning)**: reemplazar FC y softmax, fijar solo las primeras capas, aplicar *backpropagation* al resto.
 - Últimas capas suelen contener características más específicas a las categorías por las que fueron entrenadas.

<http://cs231n.github.io/transfer-learning/>

Transfer Learning en Deep Learning

image
conv-64
conv-64
maxpool
conv-128
conv-128
maxpool
conv-256
conv-256
maxpool
conv-512
conv-512
maxpool
conv-512
conv-512
maxpool
FC-4096
FC-4096
FC-1000
softmax

Red entrenada
sobre un
dataset grande
(ImageNet)



Dataset
pequeño:
Extractor de
características

Congelar
capas

Entrenar
solo esto



Dataset
mediano:
Refinado

Congelar
capas

1/100 del learning rate

Entrenar
solo esto

1/10 del learning rate

Dataset con
más datos:
Descongelar
más capas de
la red

Transfer Learning en Deep Learning

	Dataset muy similar	Dataset muy diferente
Muy pocos datos	Feature extractor con un clasificador en la última capa (ejemplo)	¡Problemas! Intenta un clasificador desde diferentes etapas (ejemplo)
Algunos datos	Finetuning con pocas capas	Finetuning con un gran número de capas

Transfer Learning en Deep Learning

- Aplicaciones:
 - Para **visión por computador**: Quizás donde más se esté aplicando esta técnica hoy en día.
 - Modelos pre-entrenados: [VGG-16](#), [VGG-19](#), [Inception V3](#), [Xception](#), [ResNet-50](#)
 - Para procesamiento de **lenguaje natural**: Aquí se hace difícil pero se puede se puede reutilizar modelos de word embedding.
 - Modelos pre-entrenados: [Word2Vec](#), [GloVe](#), [FastText](#)
 - Para **audio/habla**: modelos de reconocimiento automático del habla (ASR) desarrollados para el inglés se han usado con éxito para otros lenguajes como el alemán.