

# Bias and Variance With Mismatched Data

Train	Train-Dev	Dev	Test
-------	-----------	-----	------

학습에 사용 X

	수집, 구매한 데이터	실제 사용될 어플리케이션에서 얻은 데이터
사람 수준	사람 수준 오차	
훈련된 데이터에서 생긴 오차	훈련 오차 <i>Train</i>	회피가능 편향 avoidable bias
훈련되지 않은 데이터에서 생긴 오차	훈련-개발 오차 <i>Train-Dev</i>	분산 문제 Variance 개발/시험 오차 <i>Dev Test</i> 데이터 불일치 data mismatched

# Transfer learning

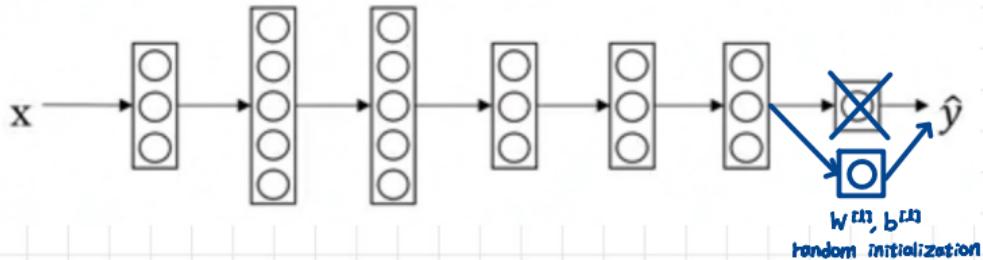


image recognition

↓ transfer

radiology diagnosis

$(x, y)$

radiology image  
diagnosis

기존 학습된 model에서 마지막 층을 제거한 후, 분류하고자 하는 문제에 적합한 층을 연결시켜 학습

- 1) 풀고자 하는 문제의 데이터가 많다면, 모든 layer 재학습
- 2) 풀고자 하는 문제의 데이터가 적다면, 새로 추가한 마지막 layer만 학습

→ 기존에 학습된 지식을 데이터가 적은 문제에 적용 가능

[Transfer learning을 사용하는 경우]  $A \rightarrow B$

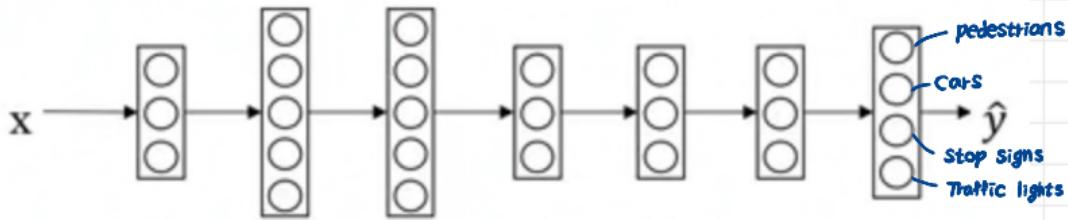
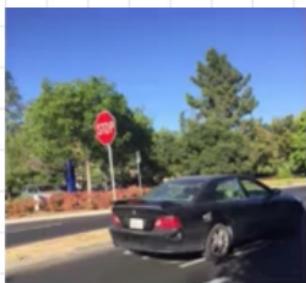
- Task A와 Task B가 같은 input, A의 low level feature가 B에 도움이 되는 경우
- Task A의 데이터 양 > Task B의 데이터 양

# Multitask Learning

하나의 신경망이 여러 task를 동시에 할 수 있도록 학습하는 것

ex) 이미지 다중 분류

신경망 초기 흑성들이 여러 물체에서 공유 가능 → multitask learning



$$\text{LOSS} : \frac{1}{m} \sum_{i=0}^m \sum_{j=0}^4 L(\hat{y}_i^{(j)}, y_i^{(j)})$$

[Multitask learning을 사용하는 경우]

- 1) 여러 task들이 하나의 low-level feature를 공유할 때
- 2) task들이 갖는 데이터의 양이 비슷할 때
- 3) 신경망의 크기가 충분히 클 때

# What is end-to-end deep learning?

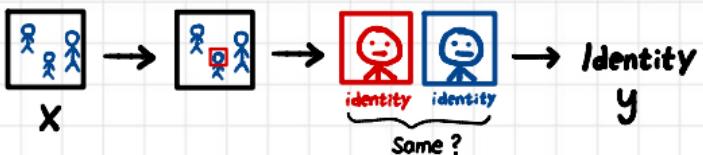
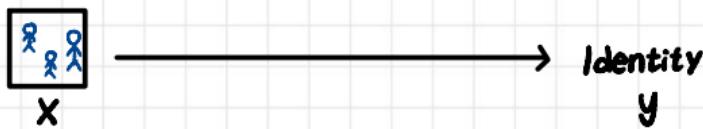
자료처리 시스템 / 학습 시스템에서 여러 단계의 필요한 과정을 한 번에 처리

[Speech recognition]



→ 엄청나게 많은 데이터가 필요

[Face recognition]



end-to-end 보다 단계를 나누는 것이 효율적일 수 있음

복잡한 문제 → 여러개의 단순한 문제

문제에 맞는 데이터 활용