|  |  |
| --- | --- |
| 교육 제목 | **딥러닝 전이학습** |
| 교육 일시 | 2021년 12월 10일 금요일 |
| 교육 장소 | 영우글로벌러닝 2층 |
| **교육 내용** | |
| 오전 | 1. 전이 학습 및 미세 조정 2. 설정 3. 소개 4. 레이어 고정: 훈련 가능한 속성 이해하기 5. 훈련 가능한 속성 의 재귀 설정    1. trainable = False    2. 기존 모델을 적용하지 않는다 6. 일반적인 전이 학습 워크플로    1. 기본 모델을 인스턴스화하고 사전 훈련된 가중치를 로드    2. trainable = False 를 설정하여 기본 모델의 모든 레이어를 고정합니다.    3. 기본 모델에서 하나(또는 여러 개) 레이어의 출력 위에 새 모델을 만듭니다.    4. 새 데이터 세트에서 새 모델을 학습시키십시오. 7. 미세 조정 8. 맞춤형 훈련 루프를 통한 전이 학습 및 미세 조정 9. 종단 간 예제: 고양이 대 개 데이터 세트에서 이미지 분류 모델 미세 조정    1. 데이터 가져오기    2. 데이터 표준화    3. 랜덤 데이터 증대 사용 10. 모델 구축 11. 최상위 레이어 훈련 12. 전체 모델의 미세 조정 라운드 수행 13. 예제     1. base\_model = keras.applications.Xception(     2. weights="imagenet", # Load weights pre-trained on ImageNet.     3. input\_shape=(150, 150, 3),     4. include\_top=False,     5. )     6. # Do not include the ImageNet classifier at the top.     7. # Freeze the base\_model     8. base\_model.trainable = False     9. # Create new model on top     10. inputs = keras.Input(shape=(150, 150, 3))     11. x = data\_augmentation(inputs) # Apply random data augmentation     12. # Pre-trained Xception weights requires that input be scaled     13. # from (0, 255) to a range of (-1., +1.), the rescaling layer     14. # outputs: `(inputs \* scale) + offset`     15. scale\_layer = keras.layers.Rescaling(scale=1 / 127.5, offset=-1)     16. x = scale\_layer(x)     17. # The base model contains batchnorm layers. We want to keep them in inference mode     18. # when we unfreeze the base model for fine-tuning, so we make sure that the     19. # base\_model is running in inference mode here.     20. x = base\_model(x, training=False)     21. x = keras.layers.GlobalAveragePooling2D()(x)     22. x = keras.layers.Dropout(0.2)(x) # Regularize with dropout     23. outputs = keras.layers.Dense(1)(x)     24. model = keras.Model(inputs, outputs)     25. model.summary() |
| 오후 | 1. 컴퓨터 비전에서 말하는 전이학습    1. 주로 사전학습 된 모델(Pre-trained model)을 이용하는 것    2. 사전 학습 모델이란 풀고자하는 문제와 비슷하면서 사이즈가 큰 데이터로 이미 학습이 되어 있는 모델 2. 왜 Transfer Learning(전이 학습)을 사용하는가?    1. 이미 학습된 모델을 사용해서 문제를 해결할 수 있습니다.    2. 이미 학습된 많은 모델은 적용하려는 데이터가 학습할 때의 데이터와 같은 분포를 가진다고 가정으로 했을 때 효율적입니다.    3. 새로운 문제를 해결하려할 때 데이터의 분포가 바뀌면 기존의 통계쩍 모델을 새로운 데이터로 다시 만들어야 할 때 좋습니다.    4. 복잡한 모델일 때 학습 시간이 오래 걸릴 수 있으며, 학습시키는데 어려움이 있습니다.    5. 층(layer)의 개수, Activation function, Hyper-parameters등 모델을 구성하는데 고려해야할 사항이 많으며, 직접 모델을 구성하여 학습시킨다는 것은 많은 시도가 필요합니다.    6. 따라서 이미 잘 훈련된 모델이 있고, 만드려는 모델과 유사한 문제를 해결하는 모델일 경우에는 Transfer Learning(전이 학습)을 사용합니다. 3. Fine-tuning 이란?    1. 기존에 학습되어져 있는 모델을 기반으로 아키텍쳐를 새로운 목적에 맞게 변형하고 이미 학습된 모델 Weights로 부터 학습을 업데이트 하는 방법을 말합니다. (pre-trained model) (다른 레이어를 고정시키고 일부분 layer를 조정)    2. 또한, 모델의 파라미터를 미세하게 조정하는 행위를 말합니다.    3. 가장 쉽게 이용하려면 내 데이터를 해당 모델로 예측(predict)하여 보틀넥 피쳐만 뽑아내고, 이를 이용하여 어파인 레이어(Fully-Connected Layer)만 학습시켜서 사용하는 방법을 취하면 됩니다. 하지만 이 경우는 피쳐를 추출해내는 레이어의 파라미터를 업데이트 하지 않기 때문에 Fine-tuning이라고 부르지 않습니다.    4. "Fine-tuning 했다" 라고 말하려면 기존에 학습이 된 레이어에 데이터를 추가로 학습시켜 파라미터를 업데이트 해야합니다. 이 때 주의할 점은, 완전히 랜덤한 초기 파라미터를 쓴다거나 가장 아래쪽의 레이어의 파라미터를 학습하게 되면 오버피팅이 일어나거나 전체 파라미터가 망가지는 문제가 생깁니다.    5. 전이학습 방안       1. Strategy 1 : 전체 모델을 새로 학습          1. 사전학습 모델의 구조만 사용하면서, 자신의 데이터셋에 맞게 전부 새로 학습시키는 방법       2. Strategy 2 : Convolutional base 일부분은 고정, 나머지 계층과 Classifier를 새로 학습          1. 낮은 레벨의 계층은 일반적인 특징(독립적인 특징)을 추출하고, 높은 레벨의 계층은 구체적이고 특유한 특징(Task에 따라 달라지는 특징)을 추출합니다. 이런 특성을 이용하여 어느 정도까지 재학습시킬지 정할 수 있습니다.          2. 데이터 셋이 작고 모델의 파라미터가 많다. -> 오버피팅이 일어날 수 있다.          3. 데이터 셋이 크고 모델이 작다. -> 더 많은 계층을 학습시켜 적합한 모델로 발전가능하다.       3. Strategy 3 : Convolutional base는 고정시키고, Classifier만 새로 학습          1. 컴퓨팅 연산 능력이 부족하거나, 데이터 셋이 너무 작을 때, 혹은 적용하려는 Task가 학습모델이 이미 학습한 데이터 셋과 매우 비슷할 때 고려해볼 수 있습니다. |