

# transfer learning

캐글 필사 스터디에서 발표한 내용

해당 회차 필사 노트북 : <https://www.kaggle.com/submarineering/submarineering-even-better-public-score-until-now>

## 전이학습 (transfer learning)

- 대량의 데이터셋으로 이미 학습이 되어있는 모델을 사용하는 것
  - pretrained model (사전학습된 모델) 이라고 함

### 장점

- 모델을 초기화하는 것보다 사전학습된 모델의 초기값을 사용하는 것이 빠르게 높은 정확도에 도달할 수 있음
- 데이터가 적어도 좋은 피처를 만들 수 있다.

### 단점

- 데이터셋이 작으면 지나치게 overfit될 수 있다.

### 사용

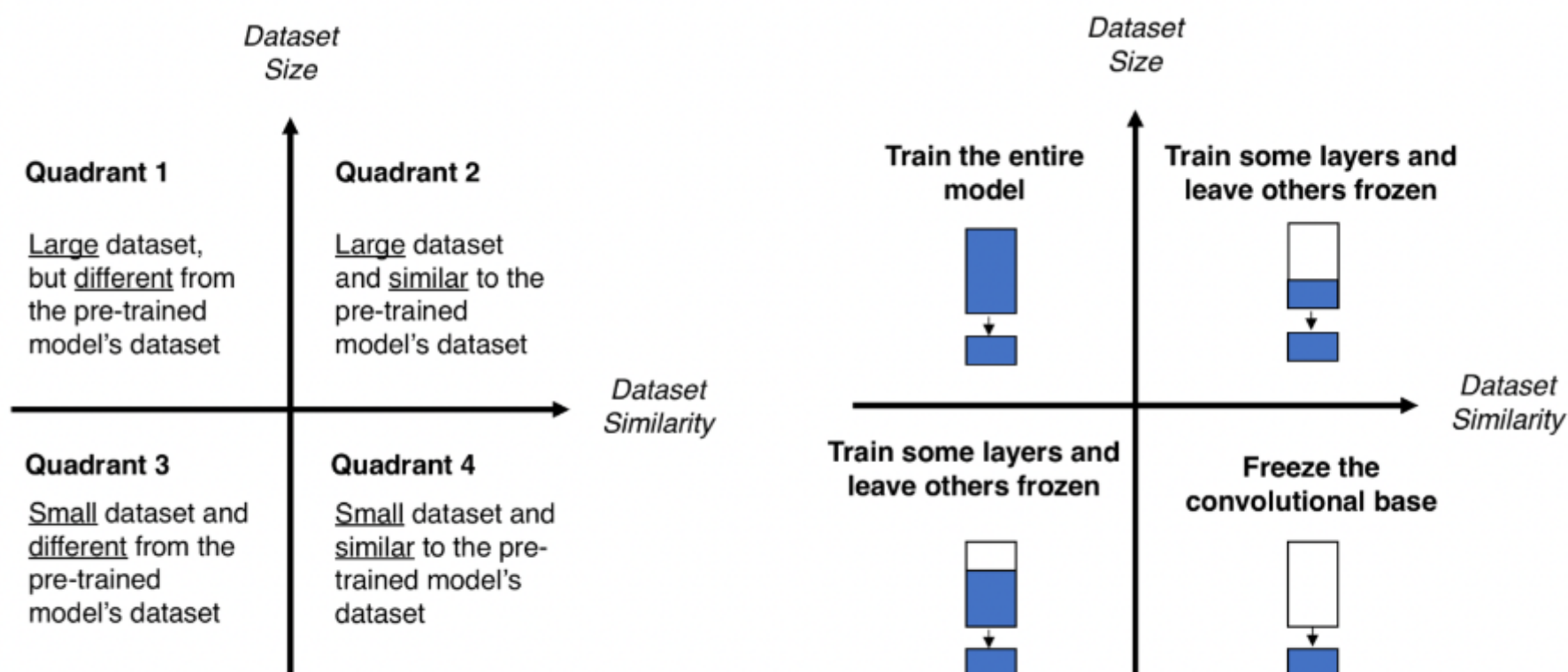


그림 3. pre-training [출처:Kaggle]

- pretrained model 도메인이 전혀 다르고, 내 데이터 셋의 양은 많다 → pretrained model 모두 프리징을 풀고 다 학습하는게 좋다.
- 데이터셋 많고 유사도 비슷 → specific한 레이어 세네개를 풀어서 학습
- 데이터셋이 적고 유사도 적음 (최악) → input레이어 제외하고 모두 재학습
- 데이터셋 적고 유사도 큰 경우 → classifier, represent만 풀어서 학습

### 코드

```
include_top=False
```

맨 아래층의 layer (fully connected layer) 를 가져오지 않는다.

⇒ 새로운 layer를 추가할 때 사용함

```
weights='imagenet'
```

imagenet으로 사전학습된 가중치를 사용한다.

- Imagenet ?
  - 대용량 데이터셋
  - 일상생활에서 볼 수있는 대부분의 이미지가 포함

```
def getVggAngleModel():
    input_2 = Input(shape=[1], name="angle")
    angle_layer = Dense(1, )(input_2)
    base_model = VGG16(weights='imagenet', include_top=False,
                        input_shape=X_train.shape[1:], classes=1)
    x = base_model.get_layer('block5_pool').output

    # layer 쌓기
    x = GlobalMaxPooling2D()(x)
    merge_one = concatenate([x, angle_layer])
    merge_one = Dense(512, activation='relu', name='fc2')(merge_one)
    merge_one = Dropout(0.3)(merge_one)
    merge_one = Dense(512, activation='relu', name='fc3')(merge_one)
    merge_one = Dropout(0.3)(merge_one)

    predictions = Dense(1, activation='sigmoid')(merge_one)

    model = Model(input=[base_model.input, input_2], output=predictions)
    sgd = SGD(lr=1e-3, decay=1e-6, momentum=0.9, nesterov=True)
    model.compile(loss='binary_crossentropy',
                  optimizer=sgd,
                  metrics=['accuracy'])

    return model
```

## Data Augmentation

### 데이터 증강

- 학습할 데이터의 양이 적을 때 사용
- 데이터에 변형을 주어 양을 늘림
- 사람이 보기엔 같은 사진이지만 컴퓨터가 볼 때는 그렇지 않음

```
gen = ImageDataGenerator(horizontal_flip = True,
                          vertical_flip = True,
                          width_shift_range = 0.,
                          height_shift_range = 0.,
                          channel_shift_range=0,
                          zoom_range = 0.2,
                          rotation_range = 10)
```