

프로젝트 #2

제 목: 시퀀스 데이터 모델 및 CNN 전이학습 구현

1. 프로젝트 자체 평가(본인 평가)

① 완성도 및 성실성(20점) : 상(20)(0) / 중(18)() / 하(16)()

2. 프로젝트 외부 평가(교수 평가)

① 과제 #1의 완성도(15점) : 상(15)() / 중(12)() / 하(10)()

② 과제 #2 완성도(35점) : 상(35)() / 중(30)() / 하(25)()

③ 보고서 작성 성실성(30점) : 상(30)() / 중(26)() / 하(22)()

④ 제출일 미준수로 인한 감점(-20점): 제출일 미준수(-20)()

소계: () 점

※ 총점: () 점

과 목 명: 패턴인식과 딥러닝

학 과: 인공지능학과

학 번: 4102211018

성 명: 연 람 희

제 출 일: 2022년 12월 8일(수)

목차

1. 목적
2. 서론
3. 네트워크 설계
4. 실험
5. 결론

1. 목적

CIFAR-100 데이터셋을 전이 학습 하는데 목적이 있다. 모델 예측 정확도 보다는 설계 방식 및 학습 방법을 어떻게 구상하였는지 문서화 하는데 목표를 둔다.

2. 서론

CIFAR-100 데이터셋은 100개의 라벨을 가진 사진 데이터셋이다. 상세하게 분류된 100개의 라벨로 분류를 진행할 수 있지만 20개의 super class가 있어 조금 더 포괄적인 분류를 할 수 있게 한다. 해당 프로젝트에서는 super class를 사용하여 분류를 진행한다.

모델 구현은 functional Keras API를 활용하였으며 데이터의 크기로 인해 google colab을 활용하였다. 전이 학습을 위해 사전 학습된 모델은 ResNet50과 MobileNetV3Large를 활용하였다.

3. 네트워크 설계

- 데이터셋 구분: train 45,000개, test: 10,000개, validation: 5,000개
- 입력 데이터: 32 x 32 x 3 이미지
- FC layer: 3 layers
- CNN layer trainable: False
- Include FC layer: False
- Epochs: 5

초기 설계로는 CNN layer와 FC layer를 기존 모델에서 호출하지 않고 직접 layer를 쌓는 방식을 선택 하였다. 전체 layer를 학습시키는 것과 FC layer를 단일적으로 학습 시키는 것을 비교하는데 목적을 두었으며 실험을 진행할수록 점차 FC layer를 증가시키고 CNN layer를 함께 학습 시키는 방법으로 진행하였다.

4. 실험

초기 실험으로 ResNet50 모델을 선택하였다. ResNet50은 마지막 FC layer층을 제외하고 23,587,712개의 파라미터를 가지고 있는 모델이다. 해당 모델에 FC layer를 3개 층으로 쌓았으며 총 26,220,948개의 파라미터로 만들었다.

```
model.summary()
```

Model: "model"

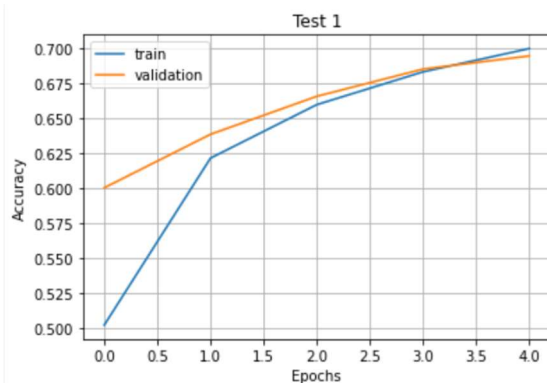
Layer (type)	Output Shape	Param #
input_3 (InputLayer)	[(None, 32, 32, 3)]	0
up_sampling2d (UpSampling2D)	(None, 224, 224, 3)	0
resnet50 (Functional)	(None, 7, 7, 2048)	23587712
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
dense (Dense)	(None, 1024)	2098176
dense_1 (Dense)	(None, 512)	524800
classification (Dense)	(None, 20)	10260

=====
 Total params: 26,220,948
 Trainable params: 2,633,236
 Non-trainable params: 23,587,712
 =====

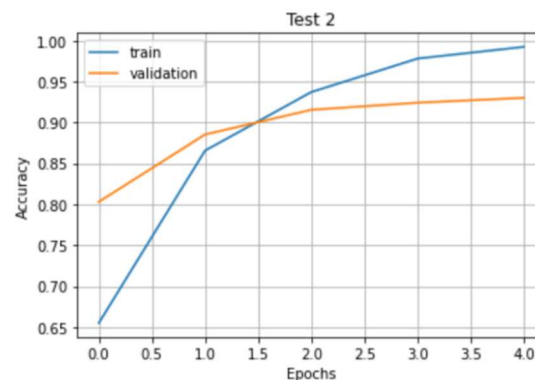
<그림1. ResNet50 model>

FC layer를 커스터마이징 하여 학습시킨 ResNet50 모델은 정확도가 67.51%로 낮은 수치를 보여주었다.

ResNet50의 정확도 향상을 위하여 기존과 같은 조건으로 CNN layer를 전부 학습시키도록 모델을 재설정 하였으며, 결과적으로 86.61%의 높은 정확도를 보여주었다.



<그림2. ResNet50 FC layer 훈련>



<그림3. ResNet50 전체 layer 훈련>

추가적으로 ResNet50 이외에도 MobilNetV3 모델을 활용하여 학습을 진행해 보았다.

해당 모델 또한 FC layer를 제외하고 훈련을 진행하였고 후에 FC layer를 추가하여 재훈련 시켰다. 다만 ResNet50과 다른 점은 해당 모델은 FC layer를 4개 층으로 구성하였고 3개층은 512개의 유닛, 마지막 층은 20개의 유닛으로 output layer를 구성 하였다.

```
model.summary()
```

Model: "model"		
Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 32, 32, 3)]	0
up_sampling2d (UpSampling2D)	(None, 224, 224, 3)	0
MobilenetV3large (Functional)	(None, 7, 7, 960)	2996352
global_average_pooling2d (GlobalAveragePooling2D)	(None, 960)	0
dense (Dense)	(None, 512)	492032
dense_1 (Dense)	(None, 512)	262656
dense_2 (Dense)	(None, 512)	262656
classification (Dense)	(None, 20)	10260
Total params: 4,023,956		
Trainable params: 1,027,604		
Non-trainable params: 2,996,352		

<그림4. MobileNetV3 FC layer 훈련>

Mobile의 프로세서에 맞게 파라미터의 수가 상대적으로 적은 모델이며 학습의 속도가 빨라 학습 진행이 수월하였다.

FC layer를 학습시킨 모델의 정확도는 63.48% 정도의 수준이었으며 다소 낮은 수치를 보여주었다. 후에 CNN layer 모두 학습시킨 모델은 81.25%의 정확도를 보여주었으며 모든 layer를 전이 학습 시켜야 학습이 잘된 모델이 완성된다는 결과를 나타내고 있다.

MobileNetV3의 경우 파라미터의 수를 조금 더 늘려 학습을 진행해 보았다. 기존 4,023,956개의 파라미터 수에서 5,565,076개로 증가시켜 layer를 구성

하였다.

```
model.summary()
```

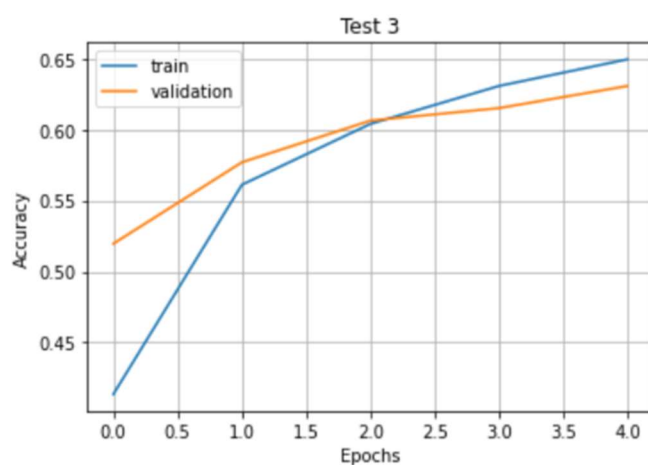
Model: "model_2"

Layer (type)	Output Shape	Param #
input_6 (InputLayer)	[(None, 32, 32, 3)]	0
up_sampling2d_2 (UpSampling 2D)	(None, 224, 224, 3)	0
MobileNetV3large (Functional)	(None, 7, 7, 960)	2996352
global_average_pooling2d_2 (GlobalAveragePooling2D)	(None, 960)	0
dense_6 (Dense)	(None, 1024)	984064
dense_7 (Dense)	(None, 1024)	1049600
dense_8 (Dense)	(None, 512)	524800
classification (Dense)	(None, 20)	10260

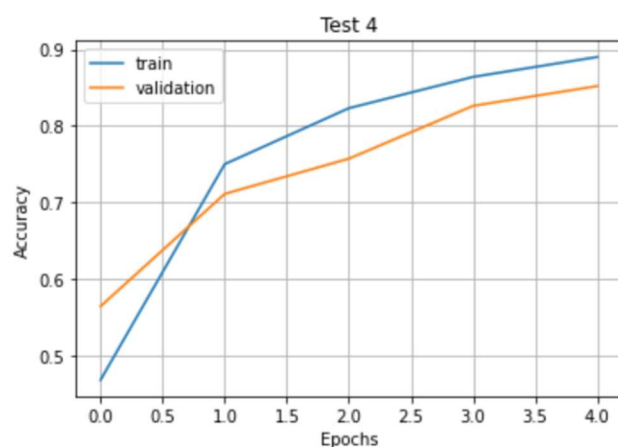
=====
 Total params: 5,565,076
 Trainable params: 5,540,676
 Non-trainable params: 24,400
 =====

<그림5. MobileNet 파라미터 수 증가 훈련>

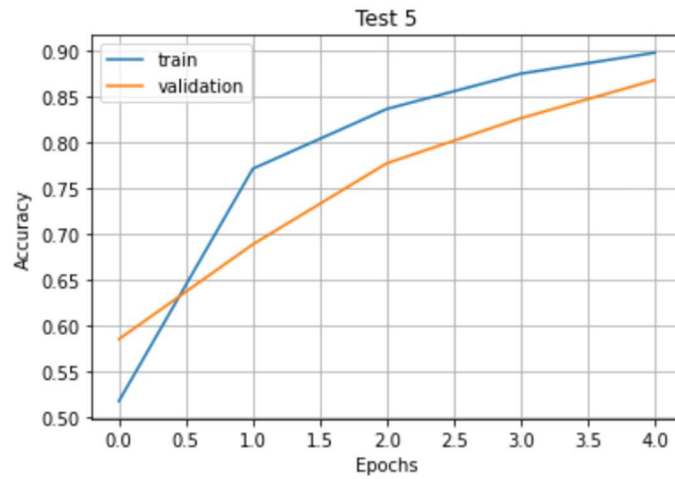
결과적으로 아주 근소한 차이였지만 82.88%의 정확도로, 이전 MobileNetV3 전이 학습 모델들에 비해 조금 더 나은 정확도를 보여주었다.



<그림6. MobileNet FC layer 훈련>



<그림7. MobileNet 전체 layer 훈련>



<그림8. MobileNet 파라미터 수 증가 후 훈련>

5. 결론

모델을 직접 설계하고 학습시키는데에는 많은 시간과 비용이 소모된다. 그러나 전이 학습은 위의 실험을 통해서도 알 수 있듯이 정확도 면에서 우수한 성능을 보여주었고 학습 소요시간 또한 줄어들어 비용을 아낄 수 있었다.

많은 pre-trained 모델들이 있지만 위의 ResNet50과 MobileNetV3를 CIFAR-100 데이터셋을 활용해 전이 학습을 실행한 결과는 아래와 같다.

	ResNet50(FC)	ResNet50(All)	MobileNetV3(FC)	MobileNetV3(All)	MobileNetV3(exp)
Parameters	26,220,948	26,220,948	4,023,956	4,023,956	5,565,076
Accuracy	0.7001	0.9922	0.6496	0.8899	0.8972
Val_accuracy	0.6948	0.9300	0.6308	0.8520	0.8673
Epochs	5	5	5	5	5
Evaluation	0.6551	0.8661	0.6348	0.8125	0.8288

<표1. All은 CNN layer 모두 학습 시킨 실험이며 exp는 파라미터수를 확장시킨 실험>

전이 학습은 실험을 통해 기존 모델의 모든 레이어를 함께 학습하는 방식이 가장 효율적이라는 것을 알 수 있었다. 또한 파라미터 수를 증가함에 따라 정확도는 점차적으로 증가하는 점을 발견할 수 있었다. 다만 MobileNetV3 모델은 모바일 환경에서 구동되는 모델이라는 점에서 파라미터의 증가는 비효율적일 수 있겠으나 단순히 파라미터 수 증가는 모델 학습에 있어서 좋은 효과가 있다는 것을 보여준다.

결과적으로 시간과 비용을 절약하는 측면에서 전이 학습을 선택하는 것은 좋은 선택이 될 수 있을 것이다.