

자비스 project

팀장 이정우
부팀장 한인혁
팀원 정주희, 최가은



01

Inception model

Inception model 소개

02

Study

공부한 함수

03

Visualization & Results

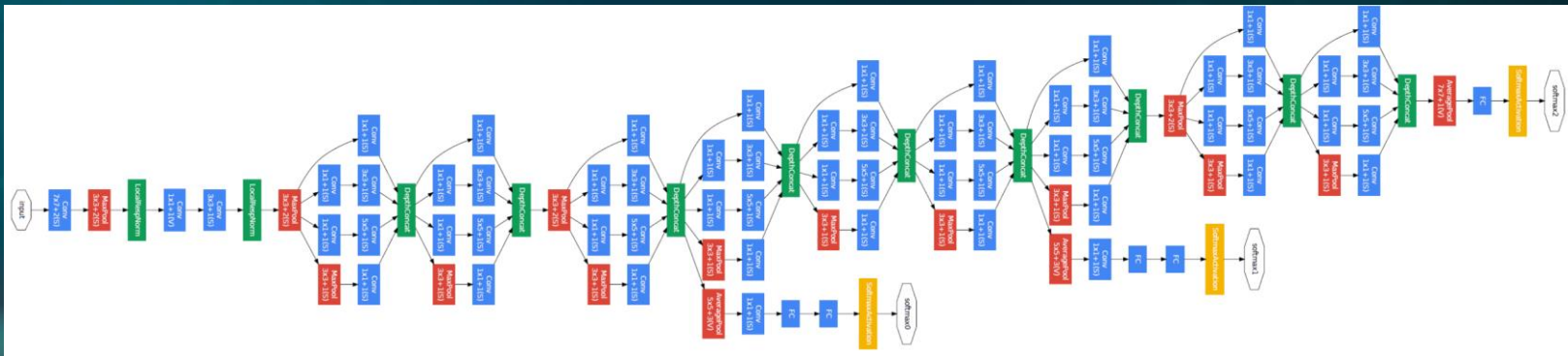
시각화 및 결과

Inception model

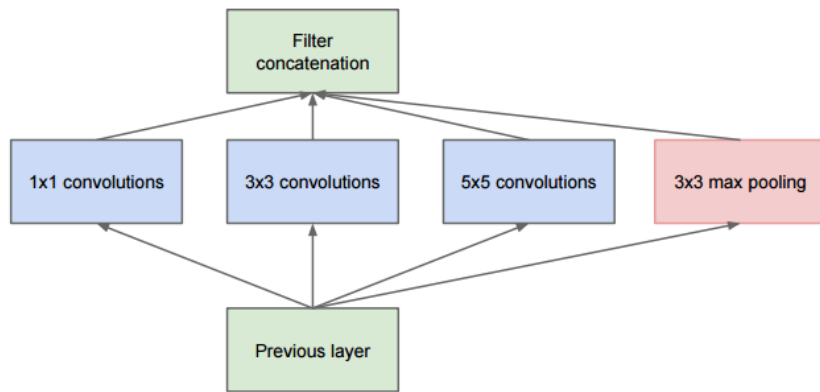
01

InceptionV1(GoogLeNet)

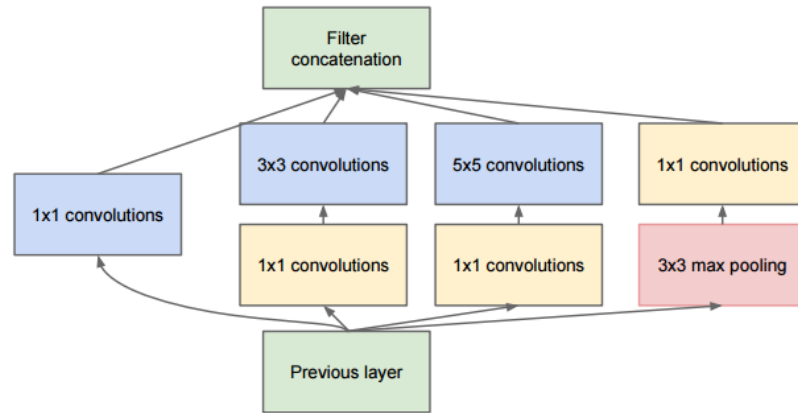
VGG19 이후 나온 모델이며 GoogleNet이라고 불리기도 하였으나 google에서 inception의 여러 버전 중 하나가 GoogleNet이며 Inception이라는 논문으로 발표하였다.



InceptionV1(GoogLeNet)



(a) Inception module, naïve version



(b) Inception module with dimension reductions

보통적인 CNN모델은 하나의 Convolution 필터로 연산하지만 Inception모델에서는 작은 Conv레이어 여러 개를 한 층에서 구현하는 형태이다.

b는 a의 개선 모델로 1x1 사이즈의 필터로 conv 해주는 것입니다.

Problem(vanishing gradient)

- 은닉층이 많은 다층 퍼셉트론에서, 은닉층을 많이 거칠수록 전달되는 가중치가 업데이트가 제대로 되지 않아 학습이 되지않는 현상
- 이 문제를 해결하기 위해 **auxiliary classifier(보조분류기)**를 달아 해결하였다.

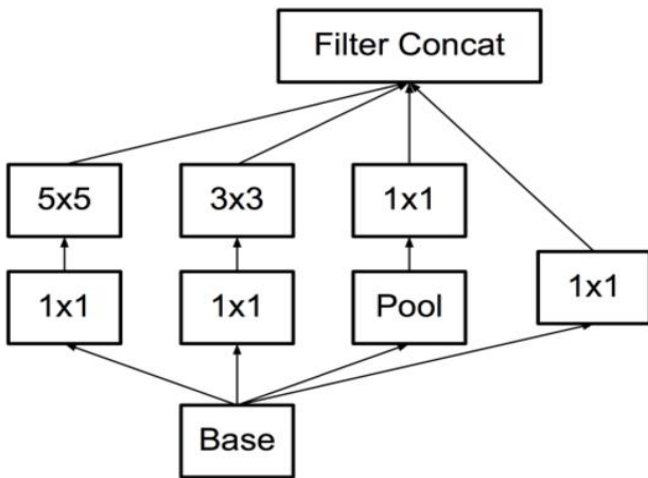
InceptionV2 / V3

Google에서 Inception V1 모델은 응용하기 쉽지 않은 모델로 판단하여 개선 시킨 모델이 Inception V2 모델이다.

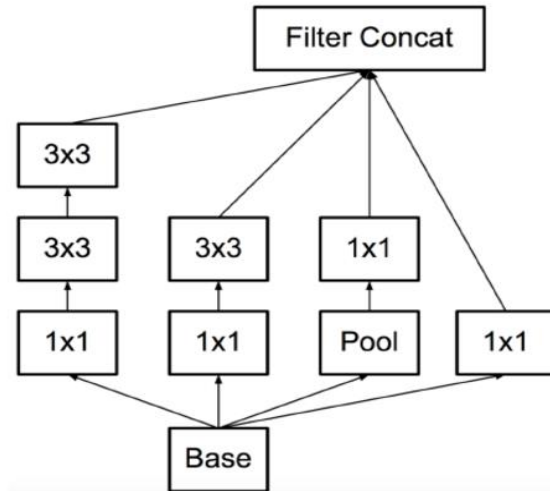
Inception V2 모델은 첫 번째 보조 분류기의 효율이 떨어져 제거하고, 모든 필터를 3x3으로 적용하였다.

Inception V3 모델은 V2 모델을 계속 수정하다 가장 좋은 결과를 묶어 만든 것이다.

InceptionV2 / V3



original inception module



factorizing inception module

$$(5*5*N) : (3*3*N) + (3*3*N) = 25 : 9+9 = 25 : 18 \text{ (약 28\% 비용 감소)}$$

Study

02

Study

```
train_datagen = ImageDataGenerator(rescale = 1./255,  
                                   shear_range = 0.2,  
                                   zoom_range = 0.2,  
                                   validation_split=0.1,  
                                   horizontal_flip = True)
```

Rescale 크기 재조절 인수 default = None

Shear_range 원본 이미지를 랜덤하게 전단변환할 각도범위

Zoom_range 랜덤하게 사진을 확대할 범위

Validation_split 학습 데이터 테스트 데이터 분류 비율 설정

Horizontal_flip 랜덤하게 이미지를 수평으로 뒤집음

Vertical_flip 랜덤하게 이미지를 수직으로 뒤집음

Fill_mode 회전이나 가로/세로 이동으로 인해 새롭게
생성해야 할 픽셀을 채울 방식

Rotation_range 랜덤하게 사진 회전시킬 범위

width_shift_range 그림을 수평으로 랜덤하게 평행이동

Height_shift_range 그림을 수직으로 랜덤하게 평행이동

Study

```
tf.keras.applications.InceptionV3(  
    include_top=True,  
    weights="imagenet",  
    input_tensor=None,  
    input_shape=None,  
    pooling=None,  
    classes=1000,  
    classifier_activation="softmax",  
)
```

include_top

마지막 계층에 완전 연결 계층을 포함할지 여부

Input_shape

선택적 이미지 크기 Include_top이 false인 경우 지정

Weights

Imagenet(imagenet에서 사전 교육)
로드할 가중치 파일의 경로이다.

Input_tensor

모델의 이미지 입력으로 사용할 선택적 텐서

Pooling

Include_top이 false인 경우 선택적 풀링모드

Classes

가중치 인수가 지정되지 않을 시 이미지를 분류할 선택적 수

classifier_activation

Include_top이 false인 경우 사용할 활성화 함수

Visualization & Results

03

Results

NO	작성자	내용	소요시간(GPU)	검증 데이터 셋 loss, accuracy	InceptionV3함수	model, epochs, batch_size, image_size	제출 최종 결과(정확도)	insight
1	이정우 선택한 코드	VGG16 (기존 제작자는 19로 명시)	21min 12s	optimizer = tf.keras.optimizers.RMSprop(lr=0.0001), loss = 'categorical_crossentropy', metrics=['categorical_accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	steps_per_epoch = 100 epochs = 10 model = VGG16 224 * 244	score = 0.44982	steps_per_epoch는 한 번의 에포크 (epoch)에서 훈련에 사용할 배치 (batch)의 개수를 지정합니다.
2	이정우 선택한 코드	InceptionV3	%time을 사용하지 않아서 20~25min추정	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 10 model = inception 224*224	score = 0.63129	345위의 높은 순위 점수로 분석하여 값을 조정하면 높은 값을 얻을 수 있을 것 같다.
3	이정우 선택한 코드	InceptionV3	1h 51min 31s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 10 model = inception 448*448	score = 0.66243	이미지 크기 증가에따른 점수 증가
4	이정우 선택한 코드	InceptionV3	2h 44min 25s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 50 model = inception 224*224	Score = 0.73241	epoch증가에 따른 큰 점수 상승
5	이정우 선택한 코드	InceptionV3	12h(43200s)	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 100 model = inception 448*448	실행시간 초과로 인해 프로그램 실행 취소	실행시간에 한계가 있는 것으로 보이며 epochs50으로 감소한뒤 실행해봐야겠다. 99epochs까지 돌릴 수 있음을 확인
6	이정우 선택한 코드	InceptionV3	7h 36min 8s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 50 model = inception 448*448	Score = 0.84083	이미지 크기와 에폭 증가로 점수증가
7	한인혁 선택한 코드	InceptionV3	7h 14min 44s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 50 model = inception 448*448	Score = 0.80151	
8	이정우 선택한 코드	InceptionV3	8h 5min 29s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 16 epochs = 50 model = inception 448*448	Score: 0.90003	batch_size감소로 정밀화로 인한 점수 증가

Results

9	최가운 선택한 코드	InceptionV3	12h(43200s)	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 16 epochs = 100 model = inception 448*448	실행시간 초과로 인해 프로그래밍 실행 취소	역시나 실행시간의 한계가 존재하였지만 448*448 이미지크기와 16 배치사이즈로는 77까지의 어복을 돌릴 수 있음을 확인하였다.	
10	이정우 선택한 코드	InceptionV3	37min 7s	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'avg'	batch_size = 32 epochs = 10 model = inception 224*224	score = 0.56670	InceptionV3 함수의 인수를 사용하여 완전연결계층 사용했을 때, 가중치를 랜덤하게 주었을 때, pooling방식을 바꾸었을 때보다 모두 기존 방식인 가중치 imagenet (imagenet에서 사전학습) 완전연결계층사용x 선택적 풀링모드 사용x 일 때 점수가 높게 나왔다.	
11	이정우 선택한 코드	InceptionV3	38min 56s	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'max'	batch_size = 32 epochs = 10 model = inception 224*224	score = 0.48596		
12	이정우 선택한 코드	InceptionV3	38min 56s	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	weights='imagenet', include_top=True	batch_size = 32 epochs = 10 model = inception 299*299	score = 0.27411		
13	이정우 선택한 코드	InceptionV3	41min 1s	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights=None include_top=False	batch_size = 32 epochs = 10 model = inception 224*224	Score: 0.20376		
14	이정우 선택한 코드	InceptionV3	9h 12min 4s	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'avg'	batch_size = 15 epochs = 50 model = inception 488*488	score = 0.72010	가장 높은 점수가 나온 부분을 뽑아 기존 방식의 epochs, batch_size, image_size 를 변경하여 적용시켜 변경해 보았으나 8번째와 비교해시간은 증가하였고 점수(정확도)는 감소하였다.	
15	한인혁 선택한 코드	InceptionV3	36min 20s	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 64 epochs = 10 model = inception 224*224	score = 0.52713		
16	최가운 선택한 코드	InceptionV3	8h 13min (29607.6s)	loss='categorical_crossentropy', optimizer=RMSprop, metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 15 epochs = 50 model = inception 448*448	score = 0.85966	수치는 가장 높은 점수가 나왔던 기존 수치로 14번째와 동일하게 맞춰주었고(이미지사이즈 제외), optimizer를 RMSprop으로 변경하여 실행해보았더니 동일수치 대비 adam 보다 높은 점수(정확도)가 나왔다.	
17	정주희	InceptionV3	2172.5s(36m 21s)	loss='categorical_crossentropy', optimizer='adam', metrics=[accuracy]	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'max'	batch_size = 32 epochs = 10 model = inception 224*224	Score: 0.52902	10번째 코드 변경. rotation_range를 추가해보았으나 점수 약간 감소 8번째 설을 통해 보이는 이미지를 보면 우리가 분석할 이미지는 회전해서 정확도를 높일만한 이미지가 아니어보임. 그래서 score감소되었다고 판단됨	train_datagen = ImageDataGenerator(rescale = 1./255, rotation_range = 20, shear_range = 0.2, zoom_range = 0.2, validation_split=0.2, horizontal_flip = True)

Results

18	정주희	InceptionV3	2153s(35m 88s)	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'max'	batch_size = 32 epochs = 10 model = inception 224*224	Score: 0.37408	10번째 코드 변경 vertical_flip를 추가해보았으나 점수 감소 8번째 실험을 통해 보이는 이미지를 보면 우리가 분석할 이미지는 전부 정방향으로 되어있기 때문에 원래대로 수직으로 뒤집을 경우 오히려 모델이 학습할 때 방해가 되었을 것으로 생각됨 그래서 score 크게 감소되었다고 판단됨	train_datagen = ImageDataGenerator(rescale = 1./255, shear_range = 0.2, zoom_range = 0.2, validation_split=0.2, horizontal_flip = True, vertical_flip = True)
19	이정우 선택한 코드	InceptionV3	33min 36s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 10 model = inception 224*224	score = 0.63437	훈련 데이터와 테스트 데이터 비율을 9 : 1 기준 대비 점수 약간 상승	validation_split = 0.1
20	이정우 선택한 코드	InceptionV3	33min 32s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 32 epochs = 10 model = inception 224*224	score = 0.63398	훈련 데이터와 테스트 데이터 비율을 9 : 1 기준 대비 점수 약간 상승	validation_split = 0.4
21	이정우 선택한 코드	InceptionV3	11h 21min 10s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False	batch_size = 16 epochs = 70 model = inception 448*448	score = 0.86505	코드 결과 epochs 따른 정확도를 보았을때 과적합현상이 일어난것으로 보이며 시각화plot에도 정확도가 중간에 값이 휘는 것이 보인다.	validation_split = 0.4
22	정주희	InceptionV3	2170.7s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'max'	batch_size = 32 epochs = 10 model = inception 224*224	score = 0.40868	10번째 코드 변경 wide_shift_range를 추가해보았으나 점수감소 8번째 실험을 통해 보이는 이미지를 보면 우리가 분석할 이미지는 수평이동을 통해 모델을 학습시킬 때 정확도를 개선할 수 있는 형태가 아니기 때문에 점수 향상이 되지 않고 오히려 감소한 것으로 보임	train_datagen = ImageDataGenerator(rescale = 1./255, width_shift_range=0.1, shear_range = 0.2, zoom_range = 0.2, validation_split=0.2, horizontal_flip = True)
23	정주희	InceptionV3	2173.3s	loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy']	input_shape=IMAGE_SIZE + [3] weights='imagenet' include_top=False pooling = 'max'	batch_size = 32 epochs = 10 model = inception 224*224	Score: 0.49058	10번째 코드 변경 height_shift_range를 추가해보았으나 점수감소 8번째 실험을 통해 보이는 이미지를 보면 우리가 분석할 이미지는 수평이동을 통해 모델을 학습시킬 때 정확도를 개선할 수 있는 형태가 아니기 때문에 점수 향상이 되지 않고 오히려 감소한 것으로 보임	train_datagen = ImageDataGenerator(rescale = 1./255, height_shift_range=0.1, shear_range = 0.2, zoom_range = 0.2, validation_split=0.2, horizontal_flip = True)

Results

No	작성자	소요시간	변경항목	Score	비고
1	이정우	20~25min	Inception V3모델 Image = 244 * 244 Epoch = 10, batch_size = 32	0.63129	최초군
2	이정우	1h 51min 31s	이미지 크기 2배 증가	0.66243	
3	이정우	12h(43200h)	이미지 크기 2배, epoch = 100	x	실행시간 초과
4	이정우	8h 5min 29s	Batch_size = 16, Epoch = 50 Image_size = 488 * 488	0.90003	최고 점수군
5	한인혁	36min 20s	Batch_size = 64	0.52713	
6	최가은	8h 13min	Batch_size = 16, Epoch = 50 Image_size = 488 * 488 Optimizer = 'RMSprop'	0.85966	기존 optimizer는 'adam'

Results

No	작성자	소요시간	변경항목	Score	비고
1	이정우	20~25min	Inut_shape = IMAGE_SIZE + [3] Weights = 'imagenet' Include_top = false	0.63129	최초군
8	이정우	37min 7s	Pooling = 'avg'	0.56670	시간 상승 점수 하락
9		38min 56s	Pooling = 'max'	0.48596	
10		38min 56s	Include_top = True Image_size = 299*299	0.27411	
11		41min 1s	Weights = None	0.20376	

Results

No	작성자	소요시간	변경항목	Score	비고
1	이정우	20~25min	Rescale = 1./255 Shear_range = 0.2 Zoom_range = 0.2 Validation_split = 0.2 Horizontal_flip = True	0.63129	최초군
12	정주희	36min 21s	Rotation_range = 20	0.52902	시간 소폭 상승 점수 하락
13		35min 88s	Vertical_flip = True	0.37408	
14		2170s	Wide_shift_range = 0.1	0.40868	모두 pooling 모드 max 설정
15		2173s	Height_shift_range = 0.1	0.49058	

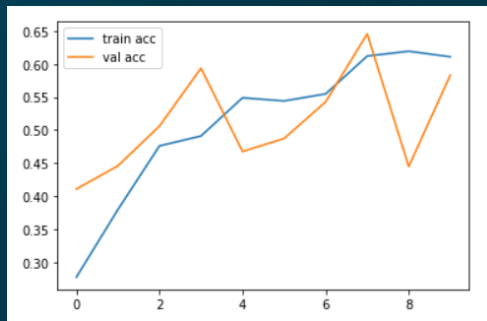
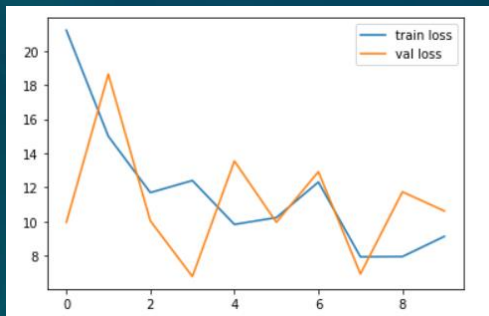
Visualization

In [18]:

```
plt.plot(r.history['loss'], label='train loss')
plt.plot(r.history['val_loss'], label='val loss')
plt.legend()
plt.show()
plt.savefig('LossVal_loss')

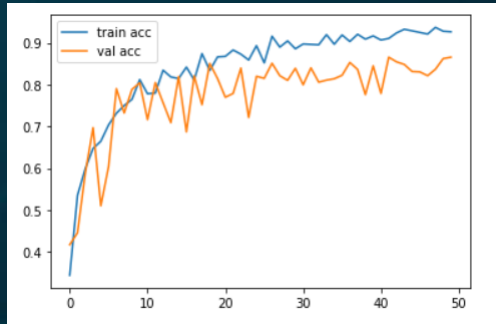
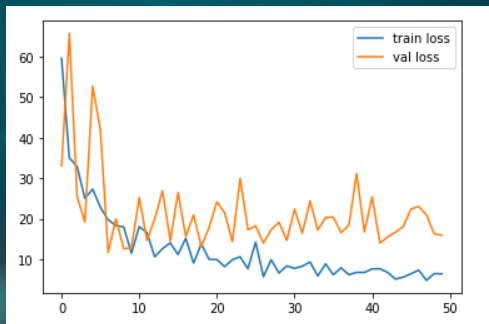
# plot the accuracy
plt.plot(r.history['accuracy'], label='train acc')
plt.plot(r.history['val_accuracy'], label='val acc')
plt.legend()
plt.show()
plt.savefig('AccVal_acc')
```

Visualization



최초실행군

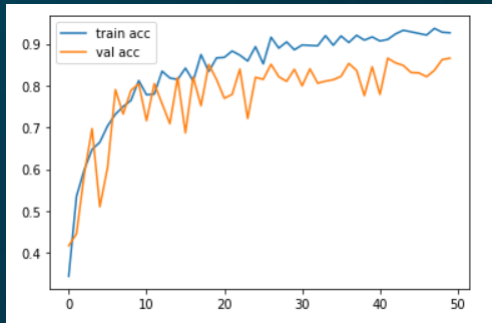
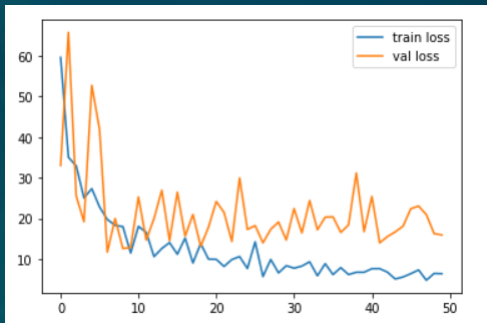
batch_size = 32
epochs = 10
model = inception
224*224
Score = 0.63129



최고점

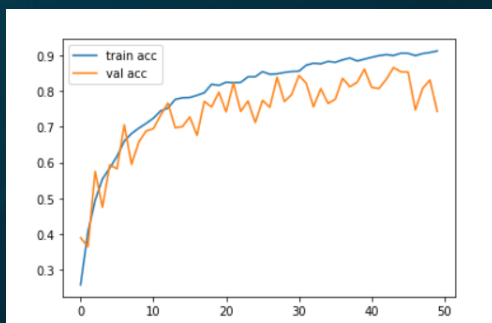
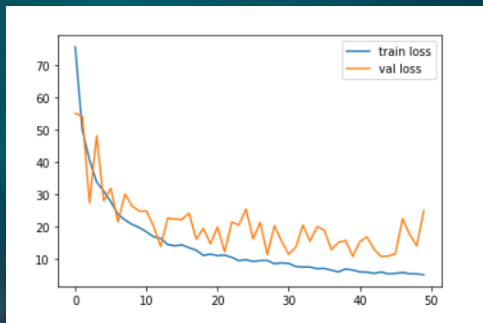
batch_size = 16
epochs = 50
model = inception
448*448
Score = 0.90003

Visualization



Adam

batch_size = 16
epochs = 50
model = inception
448*448
Score = 0.90003



RMSprop

batch_size = 15
epochs = 50
model = inception
448*448
Score = 0.85966

THANKS!

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.