

딥러닝프로그래밍 기말고사 (20241519 조예성)

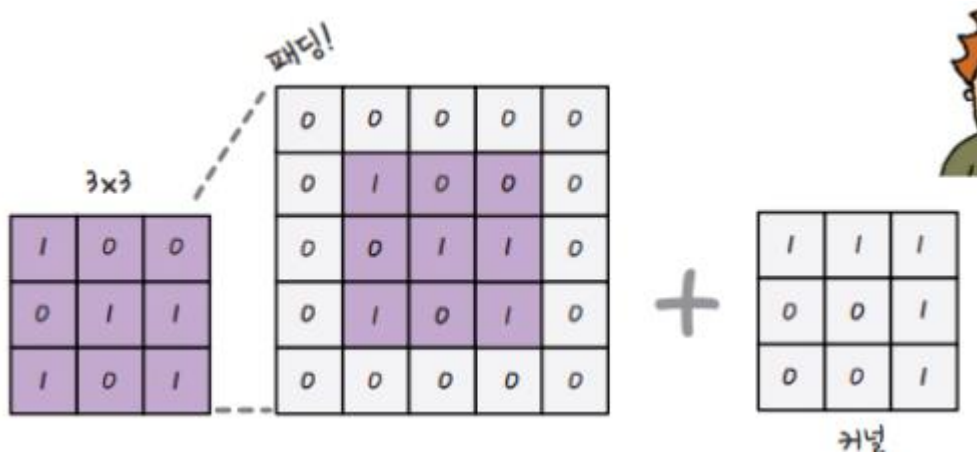
새로운 그림을 생성하는 신경망: GAN

기존의 패턴을 기반으로 그림 생성하는 것: Autoencoder

GAN 만든 사람: 이안 굿 펠로우

GAN은 (생성자)와 (판별자)로 구성됨

연산하세요



커널의 이동 간격: Stride

- 기본값: 1

```
# 모델 이름을 discriminator로 정하고 Sequential() 함수를 호출합니다.
discriminator = Sequential()
discriminator.add(Conv2D(64, kernel_size=5, strides=2, input_shape=
(28,28,1), padding="same")) ..... ❶
discriminator.add(Activation(LeakyReLU(0.2))) ..... ❷
discriminator.add(Dropout(0.3)) ..... ❸
discriminator.add(Conv2D(128, kernel_size=5, strides=2, padding="same")) ..... ❹
discriminator.add(Activation(LeakyReLU(0.2))) ..... ❺
discriminator.add(Dropout(0.3)) ..... ❻
discriminator.add(Flatten()) ..... ❼
discriminator.add(Dense(1, activation='sigmoid')) ..... ❽
discriminator.compile(loss='binary_crossentropy', optimizer='adam') ..... ❾
discriminator.trainable = False ..... ❿
```

판별자 활성화 함수: Sigmoid

```

generator = Sequential() # 모델 이름을 generator로 정하고 Sequential() 함수를 호출
generator.add(Dense(128*7*7, input_dim=100, activation=LeakyReLU(0.2))) ----- ①
generator.add(BatchNormalization()) ----- ②
generator.add(Reshape((7, 7, 128))) ----- ③
generator.add(UpSampling2D()) ----- ④
generator.add(Conv2D(64, kernel_size=5, padding='same')) ----- ⑤
generator.add(BatchNormalization()) ----- ⑥
generator.add(Activation(LeakyReLU(0.2))) ----- ⑦
generator.add(UpSampling2D()) ----- ⑧
generator.add(Conv2D(1, kernel_size=5, padding='same', activation='tanh'))
----- ⑨

```

오토인코더 사용 이유: 데이터 충분치 않을 때

생성자 모델 만들기

```
autoencoder = Sequential()
```

인코딩 부분

```

autoencoder.add(Conv2D(16, kernel_size=3, padding='same', input_shape=(28,
28,1), activation='relu')) ----- ①
autoencoder.add(MaxPooling2D(pool_size=2, padding='same')) ----- ②
autoencoder.add(Conv2D(8, kernel_size=3, activation='relu', padding='same'))
----- ③
autoencoder.add(MaxPooling2D(pool_size=2, padding='same')) ----- ④
autoencoder.add(Conv2D(8, kernel_size=3, strides=2, padding='same',
activation='relu')) ----- ⑤

```

디코딩 부분

```

autoencoder.add(Conv2D(8, kernel_size=3, padding='same',
activation='relu')) ----- ⑥
autoencoder.add(UpSampling2D()) ----- ⑦
autoencoder.add(Conv2D(8, kernel_size=3, padding='same',
activation='relu')) ----- ⑧
autoencoder.add(UpSampling2D()) ----- ⑨
autoencoder.add(Conv2D(16, kernel_size=3, activation='relu')) ----- ⑩
autoencoder.add(UpSampling2D()) ----- ⑪
autoencoder.add(Conv2D(1, kernel_size=3, padding='same',
activation='sigmoid')) ----- ⑫

```

전체 구조 확인

```
autoencoder.summary() ----- ⑬
```

미리 학습한 모델을 가져와서 사용하는 방법: 전이학습

```
transfer_model = VGG16(weights='imagenet', include_top=False, input_shape=(150,150,3))
transfer_model.trainable = False
transfer_model.summary()
```

Total params: 14,714,688

Trainable params: 0

Non-trainable params: 14,714,688

```
finetune_model = models.Sequential()
finetune_model.add(transfer_model)
finetune_model.add(Flatten())
finetune_model.add(Dense(64))
finetune_model.add(Activation('relu'))
finetune_model.add(Dropout(0.5))
finetune_model.add(Dense(1))
finetune_model.add(Activation('sigmoid'))
finetune_model.summary()
```

설명 가능한 AI 제안한 학자: 리처드 파인만

CNN 제작자: 얀 르쿤

인공지능의 결과를 알 수 없는 모델: 블랙박스 모델

설명가능한 AI 약자: XAI

중간 과정에서 나온 특징 맵 활용: CAM, Class Activation Map

```
explainer = GradCAM()
output = explainer.explain(데이터, 모델, 클래스)
explainer.save(output, 저장될 폴더, 저장될 이름)
```

대부분의 인공지능망은 (블랙박스) 모델이다

CNN에서 부위별로 가려가며 설명가능한 모델 만드는 것: 오클루전, 마스크

두가지 XAI 기법: 오클루전 CAM