

## 0605 머신러닝 대본 – 김찬영, 허진환

1page

안녕하세요 이번 머신러닝 이미지 노이즈 제거 클래스 분류기 프로젝트의 발표를 맡은 팀 셀프러닝의 팀장 김찬영과 팀원 허진환입니다. 발표 시작하겠습니다

2page

발표의 순서는 화면과 같이 설계 과제 소개, 오토인코더, 합성곱 신경망, 프로젝트 진행 현황 순으로 진행하도록 하겠습니다.

3page

첫 번째 설계과제 소개입니다. 저희가 구현하고자 하는 모델의 동작을 간단하게 표현하였습니다. 결국 본 프로젝트에서 구현하고자 하는 모델은 노이즈가 섞인 이미지의 분류입니다. 이를 하기 위해 먼저 이미지 데이터셋을 불러온 다음, 이 데이터셋에 임의의 노이즈를 발생시킵니다. 그 후 오토인코더를 사용해 최대한 원본 이미지와 비슷하도록 노이즈 제거를 합니다. 마지막으로 노이즈가 제거된 이미지를 분류하여 결과를 확인합니다. 이 모델의 성능을 개선시키는 것 또한 저희의 목표입니다.

4page

다음은 오토인코더에 대한 설명입니다. 오토인코더는 입력층과 출력층으로 구성된 신경망 모델로, 데이터의 특징 추출 및 차원 축소에 사용됩니다. 중간의 은닉층은 일반적으로 입력층보다 적은 뉴런을 가집니다.

오토인코더는 입력 데이터를 압축하기 위해 은닉층으로 정보를 인코딩하고, 이 과정에서 은닉층은 입력층보다 적은 뉴런을 가지고 있기 때문에 정보의 손실이 일어나게 됩니다. 이를 통하여 입력 데이터의 특징이 추출되고 압축된 표현이 만들어집니다.

그 후 인코딩된 데이터를 출력층으로 전달합니다. 출력층에서는 은닉층에서 압축된 표현을 다시 원래의 입력 데이터로 복원하는 프로세스가 진행됩니다. 디코딩 과정에서 입력 데이터와 최대한 비슷한 출력이 생성되는데, 복원된 출력과 원래 데이터의 차이를 최소화하는 방향으로 학습합니다. 학습을 통해 오토인코더는 입력 데이터의 특징을 학습하고 재구성하는 능력을 향상시킵니다.

5page

다음은 합성곱 신경망 CNN 에 대한 설명입니다. CNN 은 주로 이미지와 같은 그리드 형태의 데이터를 처리하는 데 특화된 신경망 구조입니다. 이미지의 지역적인 구조와 특징을 활용하여 분류 작업을 수행하기 때문에 이미지 인식이나 분류에 효과적일 것으로 예상하여 본 프로젝트에서 사용하기로 결정하였습니다. 이제 CNN 의 구조와 동작에 대해 설명해드리겠습니다.

합성곱 층은 입력 데이터로부터 이미지의 특징을 추출하는 역할을 합니다. 각 합성곱 층은 여러 개의 필터 또는 커널로 구성되어 있습니다. 필터는 입력 데이터에 대해 지역적인 패턴을 탐지하는 작은 윈도우로, 입력 데이터를 윈도우마다 합성곱 연산을 수행하여 특징 맵을 생성합니다.

합성곱 층의 출력은 비선형성을 도입하기 위해 활성화 함수를 통과하는데, 주로 ReLU(Rectified Linear Unit) 함수가 사용됩니다. ReLU 함수는 음수를 0 으로 만들고, 양수는 그대로 유지하는 함수입니다.

풀링 층은 특징 맵의 크기를 줄이거나 중요 정보를 강조하는 역할을 합니다. 일반적으로 최대 풀링이 사용되고, 최대 풀링은 윈도우 내에서 가장 큰 값을 선택하여 특징을 강조하고, 나머지 값은 버리는 방식으로 동작합니다.

완전 연결 층은 추출된 특징을 기반으로 클래스의 예측을 수행합니다. 풀링 층의 출력을 일렬로 펼친 후 완전 연결 층에 전달합니다. 완전 연결 층은 일반적인 다층 퍼셉트론 구조로 구성되어 있으며, 출력층은 클래스 분류를 위한 뉴런 수와 활성화 함수를 설정하여 최종 예측을 수행합니다.

CNN 은 합성곱층과 풀링층을 거치면서 입력 이미지의 주요 특성 벡터를 추출하여, 완전연결층을 거치면서 1 차원 벡터로 변환되고, 마지막으로 출력층에서 활성화 함수인 소프트맥스 함수를 사용하여 최종 결과가 출력됩니다.

6page

다음으로 저희 프로젝트 진행 현황에 대해 저희가 지금까지 완성한 코드를 기반으로 설명드리겠습니다.

저희가 만들고자 하는 모델이 노이즈 제거를 거쳐 결국 클래스 분류까지 이어져야 하므로 가져오는 이미지 데이터셋 또한 다양한 카테고리를 가지는 cifar10 데이터셋을 위 코드와 같이 가져왔습니다.

함수를 통해 반환받은 값을 훈련용/검증용으로 나누어 저장하고, 아래와 같이 가져온 이미지를 랜덤하게 5 곱하기 5 형식으로 출력하였습니다.

7page

가져온 이미지 데이터를 통해 노이즈 데이터를 생성해야 하므로, 보시는 코드와 같이 노이즈 강도를 0.3 으로 설정하고, 각 이미지 데이터 픽셀 값에 노이즈를 추가하여 이미지 클리핑 후 noise\_dataset 배열에 훈련용 이미지 데이터에 노이즈를 추가한 데이터를 추가하였습니다.

보시는 화면에는 나타나있지 않지만, 검증용 데이터 X\_test 에도 똑같은 방식으로 노이즈를 추가하여 noise\_test\_set 배열에 노이즈 데이터를 추가하였습니다.

8page

다음으로 학습을 시행할 오토인코더 모델을 생성하였습니다. 방금 김찬영 학우가 설명했던 것처럼 오토인코더는 데이터의 특징을 추출하고 차원을 축소하는 역할을 수행합니다. 오토인코더 역시 합성 신경망 CNN 을 기반으로 한 모델이므로, 위에 띄워드린 cnn 구조를 따르게 됩니다.

작성된 오토인코더 모델의 인코더 부분에서 먼저 Cov2D 함수를 통해 첫 번째 인자 32 로 합성곱층의 출력 필터 개수를 설정하였습니다. 각각의 필터는 작은 윈도우로서 입력 데이터에 대해 지역적인 패턴을 탐지하고, 윈도우마다 합성곱 연산을 수행해 이미지 데이터의 특성을 추출합니다. 이 때, 윈도우의 크기를 두 번째 인자인 3 컴마 3 으로 지정하였습니다. 또한, 합성곱층의 출력에 비선형성을 도입하기 위해 활성화함수인 relu 를 사용하고, 입력과 출력의 크기를 동일하게 하기 위해 padding 에 same 을 주었습니다.

이후, 풀링층으로 진입하기 위해 MaxPooling2D 함수를 사용해 2x2 최대 풀링을 넣어주어 데이터의 특징을 강조하고, 나머지 값을 버립니다. 인코더 부분은 이 풀링층에서 다음의 합성곱층으로 진입하는 차원 축소 및 압축의 과정이 작성되었습니다.

디코더 부분은 위의 역순으로 UpSampling2D(2, 2)함수를 사용하여 입력 데이터의 각 차원을 2 배로 확장하고 이미지 특성을 3 개의 필터로 추출하여 차원 복원 및 압축으로부터의 복원을 수행하도록 작성되었습니다.

9page

마지막으로, 설계한 오토 인코더 모델을 통해 데이터 훈련을 fit()함수로 진행하였고, 이는 noise\_dataset 입력에 대해 정답 레이블인 원본 데이터 X\_train 과 비교하여 학습하도록 합니다. 여기서 에포 chs 옵션을 10 으로 주어 학습 반복 횟수를 10 회로 지정하였습니다.

이후, 정확도 파라미터로 모델 성능을 측정하고, 잡음제거 결과를 맨 밑에 보시는 화면과 같이 출력하였습니다. 학습 반복 횟수를 정할 때 학습 시간과 노이즈 제거 정도, 정확도 파라미터 결과에 따라 값을 바꿔가면서 학습시켰고, 저희가 찾은 최적의 학습 횟수가 10 회라고 생각되어 보여드리게 되었습니다.

저희의 다음 단계는 클래스 분류기로, 이렇게 잡음이 제거된 이미지 데이터를 다시 로드하여 클래스분류기에 넣고 자동차 사진이면 자동차 카테고리에 들어가고, 비행기 사진이면 비행기 카테고리에 들어가는 등의 각 이미지 데이터를 카테고리에 따라 분류하는 단계를 준비하고 있습니다.

10page

이상으로 발표 마무리하고 질문 받겠습니다. 감사합니다.