**머신러닝 팀 프로젝트**

|  |  |
| --- | --- |
| 프로젝트 주제 | U\_Net과 GAN을 사용해서 흑백 이미지 색칠하기 |
| 팀 명 | 런닝머신 |
| 팀 원 | 팀장: 주형록(20190946)  팀원: 신다운(20210927), 박지연(20210915) |
| 프로젝트 내용 | **프로젝트 목적**  - 프로젝트를 하면서 U\_Net과 GAN을 활용함으로써 이해도를 높이고, 머신러닝에 대해 더 심층적으로 다가간다. 맞는 카테고리에 분류하는 U\_Net과 소음을 섞어 결과물의 성능을 높이기 위해 훈련하는 GAN을 사용하여 흑백 이미지를 컬러화한다.  **프로젝트 내용**  - 라이브러리 설정  torch 딥 러닝 프레인워크를 사용한다. 이것을 사용하기 위해서는 신경망을 구축 및 훈  련하고, 데이터 변환 수행과 data set, data loader를 사용하는 기능이 필요하다.  - 파일 접근( 이미지 사용을 위해)  구글 드라이브 파일 시스템을 /context/drive 경로에 마운트 하여 지정된 파일에 파일  접근을 할 수 있게 해준다.  구글 드라이브에 이미지가 저장된 경로에 확장명이 jpg인 파일 이름을 찾고, 이미지의  1000개의 틀을 가져와 임의로 요소를 선택한다.  - 훈련용과 검증용 분류  1000개의 임의의 값을 생성하고 800개의 훈련용과 200개 검증용으로 배열로 나눈다.  - 이미지 크기 설정  검증용 이미지 데이터를 필요한 크기로 만든다.  SIZE x SIZE 크기로 조정하고 bicubic interpolation(바이큐빅 보관법)과 horizontal flips을  사용하여 데이터를 확대한다.  \_getitem\_\_으로 경로에 있는 이미지를 RGB로 변환하고, tensor을 사용해 명도를 추출해  50으로 나눠 1과 -1사이의 값을 조정한다.  - 흑백 이미지 컬러화하여 출력하는 과정  make\_dataloader 함수를 사용해 이미지 로드, 사전 처리 lab 색 공간으로의 변환을 수  행한다.  U\_Net을 통해 이미지 분할 작업을 한다.  patchDiscrimcator-판별기(GAN)를 통해 이미지를 입력으로 사용하여 각 값이 패치가  실제 또는 가짜일 확률을 나타내는 단일 채널을 출력하고, 네트워크를 훈련하기 위한  적대적 손실을 계산한다.  이미지 색상화를 위해서 GAN의 생성기와 판별기를 통해 모델을 훈련해서 손실 계산  한다.  colorzation 모델을 훈련한 이미지와 손실값을 출력하고 성능에 대한 값을 계산한다.  이 모든 과정을 여러번 반복하여 이미지를 출력한다. |
| 팀원간 역할  분담 | - 신다운 : 제안서, ppt, 최종 보고서 및 코드 작성, 최종 보고서 발표  - 박지연 : 제안서, ppt, 최종 보고서 및 코드 작성, 최종 보고서 발표  - 주형록 : 팀장, 제안서, ppt, 최종 보고서 및 코드 작성, 제안서 발표 |
| 프로젝트 수행방향  **(수행 방법/도구)** | - 수행도구 : 코랩 사용  코랩은 클라우드 기반의 주피터 노트북 개발 환경이다. 머신러닝은 컴퓨터 사양이 중요한데, 구글 코랩은 CPU와 램을 제공해주기 때문에 컴퓨터 성능과 상관없이 프로그램을 실습할 수 있다. 그리고 코랩에서 사용할 프로그래밍 언어는 파이썬이다.  - 수행방법 : U-Net과 GAN을 활용하여 프로젝트 진행  U-Net은 이미지 분할을 목적으로 제안된 End to End 방식의 Fully Convolutional Network 기반 모델이다. 그리고 GAN(생성적 적대 신경망)이란 비지도학습에 사용되는 머신러닝 프레임워크의 한 종류이다. 이 두가지 알고리즘을 활용해 흑백사진을 컬러사진으로 복원하는 머신러닝을 수행 할 예정이다. |
| 프로젝트  중요  사용 모델 | ● U\_Net  1. 점진적으로 넓은 범위의 이미지 픽셀을 보며 의미있는 정보를 추출하는 수축 경로  2. 해당 정보를 픽셀 위치정보와 결합하여 각 픽셀마다 어떤 객체에 속하는지 구분하는 확장 경로  3. 수축 경로에서 확장 경로로 전환되는 전환 구간  이렇게 U\_Net 모델을 세가지로 분류할 수 있다.  모델의 입력이 RGB 데이터일 때 수축경로에서는 downsampling 과정을 반복하여 특징맵을 생성하고, 주변 픽셀들을 참조하는 범위를 넓혀가며 이미지로부터 의미있는 정보를 추출한 이후 전환 구간에서는 해당 정보를 일반화하고 노이즈에 견고하게 만든다. 그이후에 확장 경로에서는 upsamling과정을 반복하여 특징맵을 생성한다. 이때 정보들은 위치정보와 결합하는데 그 이유는 여러 번의 layer를 지나면서 특징맵의 크기가 줄어들었기 때문이다. 따라서 U\_Net은 해당 카테고리를 효과적으로 분리하여 학습하기 좋은 방법이다.  ● GAN  GAN 모델은 생성자와 판별자 이렇게 두 개의 주요 구성 요소로 이루어져 있다. 생성자는 가짜 데이터를 생성하려고 노력하고, 판별자는 실제 데이터와 생성자가 생성한 가짜 데이터를 구분하려고 노력한다. 이러한 두 신경망은 경쟁 관계에 있으며, 서로를 발전시키는 방식으로 학습된다.  GAN의 작동되는 방식은 다음과 같다.  1. 생성자 : 생성자는 잠재 벡터를 입력으로 받아 실제 데이터와 유사한 가짜 데이터를 생성한다. 잠재 벡터는 일련의 난수로 구성된 벡터이며, 생성자는 이를 실제 데이터 분포와 일치시키는 방향으로 학습한다.  2. 판별자 : 판별자는 실제 데이터와 생성자가 생성한 가짜 데이터를 입력으로 받아 이를 구분한다. 판별자는 이진 분류 문제로 볼 수 있으며, 입력이 실제 데이터인지 가짜 데이터인지를 판단하는 역할을 수행한다.  3. 학습 과정 : 생성자와 판별자는 번갈아 가며 학습하는데, 생성자는 판별자를 속이기 위해 더 진짜 같은 가짜 데이터를 생성하려고 노력하고, 판별자는 생성자가 생성한 가짜 데이터를 실제 데이터와 구분할 수 있도록 학습한다. 이 과정은 두 신경망의 손실 함수를 최적화하는 경사 하강법을 사용하여 이루어진다.  이런 방식으로 GAN은 주로 이미지, 음성, 텍스트 등의 데이터를 생성하는 데 사용된다. |
| 기능  블록도 |  |
| 프로젝트 관리 방안 | - google colab 활용 : 프로젝트 관리 |
| 상세 일정 | - 6.05 : 제안서 작성/발표  - 6.05 ~ 6.09 : 프로젝트 구현, 코드 추가  - 6.09 ~ 6.11 : 최종 코드 구현, 보고서 작성  - 6.11 ~ 6.12 : 최종발표 |
| 기대 효과 | 팀 프로젝트를 위한 기본적인 머신러닝 알고리즘에 대해 이해하고 머신러닝 과목에 대한 이해도를 향상 시킨다. 또한 프로젝트에 활용된 U\_net과 GNA를 학습한뒤 흑백이미지 색칠하기를 통해 머신러닝 구현과정을 적용시킴으로써 상호간의 협동심, 협업 능력을 향상시킨다. |
| **회의록** | 밑에 첨부 |