**작업 수행 보고서**

|  |  |
| --- | --- |
| **작 업 명** | 인공지능 커플 예측 모델 개발 |
| **작업 일시** | 2022. 04. 17 ~ 04. 21 |
| **작 업 자** | 민재홍 |
| **작업 목적** | * 인공지능 모델을 통해 남, 여 각각의 사진으로 커플 확률을 자동 예측 * 1차 목표   + 추후 서비스 활용을 위해 위 목적이 구현 가능한지에 대한 검토 |
| **작업 환경** | * Google Colab 으로 구축(2022.04.21 기준 Colab으로 코드 실행 시, 자동 구축됨)   + OS platform : Linux Ubuntu 18.04 64Bit   + Script : Jupyter Notebook   + Python : 3.7.13   + 주요 Library     1. pandas : 1.3.5     2. numpy : 1.21.6     3. cv2 : 4.1.2     4. skimage : 0.16.3     5. sklearn : 1.0.2     6. tensorflow / keras : 2.8.0   + Google Drive에 접근 필요(Google 계정 필요) |
| **작업 내역** | 1. **Data Load**    1. 사진 데이터는 아래와 같이 실행 코드(couple\_classification.ipynb)와 같은 경로의 ‘01\_Couple’, ‘02\_Non\_Couple’ 폴더로 구분하여 저장함 (폴더 명을 변경할 경우, 코드 내에 반드시 수정 필요)    2. 각 ‘01\_Couple’, ‘02\_Non\_Couple’ 폴더는 아래와 같이 구분되어 있음 **01\_Couple**  **02\_Non\_Couple**    3. 위 폴더 내 아래와 같이 학습에 사용할 사진이 저장되어 있음    4. 코드에서는 아래와 같이 경로를 저장함    5. ‘couple’, ‘male’, ‘female’을 컬럼으로 하는 데이터 프레임 생성 couple : 각 남/여 사진이 커플인지 아닌지 여부(label) male : 남 사진경로 female : 여 사진경로 2. **Hyper-parameter setting**    1. Image 처리 관련, 딥러닝 학습 관련 hyper-parameter 각각 세팅    2. 사진 크기의 경우 (53 x 63) or (28 x 28) 등 각각이 다른 경우가 존재하므로 (40 x 40)으로 통일하기 위해 resize 진행(Data preprocessing) 3. **Data preprocessing**    1. 1에서 만든 데이터 프레임의 각 경로를 가져와서 해당 경로에 해당하는 이미지를 읽은 후, RGB값의 수치 배열로 변환 및 (40 x 40) resize    2. 이미지는 모두 컬러로 이루어져 있으므로, RGB값 3채널을 가짐 따라서, 하나의 이미지는 (40 x 40 x 3) 배열로 이루어짐    3. 각 이미지는 딥러닝 학습의 효율성을 증대시키기 위해, 0~1사이 값으로  Normalization 진행    4. 최종 생성된 배열은 (데이터 수 x 40 x 40 x 3)의 크기를 지님    5. 학습을 위해 남자 이미지, 여자 이미지로 구분하여 저장 4. **Show image**    1. 3의 과정이 잘 이루어졌는지 확인하기 위해, 직접 커플 or 비커플 중  랜덤하게 한 쌍을 추출하여 시각화 5. **Split train and test set**     1. 모델은 남/여를 각 쌍으로 학습해야 하므로 남/여 각각을 쌍으로 묶어 split해야 함    2. Train : Test = 8 : 2 의 비율로 저장    3. 이후, split 된 데이터를 다시 남/여로 분리 6. **Training CNN Model**    1. 모델 학습 과정으로, 모델은 이미지 인식을 위한 CNN(Convolution Neural Network) 모델 사용        1. CNN은 이미지를 인식하기 위해 패턴을 찾는데 매우 유용함. 데이터를 통해 특징(Feature map)을 스스로 학습하기 때문에 사람이 특징을 찾아서 넣어줄 필요가 없음. 이미지의 특정 위치에서 학습한 파라미터를 이용하여 다른 위치에 있는 동일한 특징을 추출할 수 있음       2. Architecture       3. 전처리된 남성, 여성이미지 각 한장씩을 모델에 주입하면, 첫 번째 convolution layer를 거치면서 feature map을 추출하게 되고, 이후 batch\_normalization layer를 거치면서 covariate shift를 막아주게 됨. 다음 Leaky RaLU 활성 함수로 비선형 변환을 거치고, Maxpooling layer를 통해 대표 값만을 추출하여 이미지 사이즈를 줄여줌. 그 다음으로 dropout을 통해 과적합을 방지하기 위한 작업 진행. 이 과정을 한 번 더 반복한 이후, concatenate + flatten을 거쳐 일반 neural net 층을 통해 확률 계산을 하게 되어 최종 커플 확률을 추출하게 됨    2. Batchsize : 8, epoch : 600, learning\_rate : 0.00001, optimizer : Adam,  validation\_split : 0.1 로 진행        1. 학습 데이터 수가 매우 적으므로 Batchsize를 작게 하여 여러 번 업데이트 되도록 진행       2. 학습 데이터 수가 매우 적으므로 loss의 발산 가능성이 존재함.  따라서 learning\_rate는 매우 적은 값으로 진행 7. **Model validation**    1. 학습된 모델의 성능을 평가하기 위한 파트로, 위에서 분리한 Test set에 대해 55% 정도의 정확도를 보이고 있음(데이터 양 문제로 인해 정확도는 낮을 수 밖에 없음)    2. 학습 진행 중 생성된 train/validation loss의 그래프 확인        1. 데이터가 매우 적은 상황임으로 아래 그래프에서 validation loss가 global optima를 찾지 못하는 것 확인 가능       2. train loss는 변동 폭이 심하나 지속적으로 줄어드는 패턴인 것으로, 모델이 학습은 가능할 것으로 판단 8. **Classification for real data(Test)**    1. 실제 서비스를 하게 될 경우, 거치게 될 과정을 Prototype으로 생성 |
| **이슈 사항** | 1. **데이터 문제**    1. 딥러닝은 안정적으로 학습하기 위해, 패턴을 정확히 구분할 수 있는 수천 ~ 수만 단위 데이터가 필요(현재 Label 당 남,여 각 50개 뿐)    2. 사진의 사이즈가 서로 다름(53 x 63), (28 x 28) 존재    3. 사진이 조금 더 선명도가 높은 경우 학습이 용이할 수 있음 2. **모델 문제**    1. 데이터가 매우 적다보니 심각한 과적합 발생    2. 이 상황에선 layer를 더 복잡하게 쌓거나, hyper-parameter를 조정해도 큰 의미가 없음(layer가 깊어지면 학습이 안되거나 오버피팅이 더 심해지는 것 확인)    3. Train loss는 변동 폭이 심하나 지속 줄어드는 패턴으로, 모델이 학습은 가능함을 보여줌. 그러나, 서로 선호하는 얼굴 보다는 같이 찍은 사진의 주변 배경과 사진의 전체적 유사함을 보고 모델이 판단했을 가능성이 높음 |
| **결 과 물** | * 코드(couple\_classification.ipynb) * 최종 학습 모델(weights.best.hdf5) |
| **기 타** | * 참조 <https://medium.datadriveninvestor.com/dual-input-cnn-with-keras-1e6d458cd979> https://towardsdatascience.com/building-a-convolutional-neural-network-cnn-in-keras-329fbbadc5f5 |