**기수 :** 13기 **이름 :** 이승현

|  |
| --- |
| A blue and white logo  Description automatically generatedA blue and white logo  Description automatically generated**25-1 DSL 정규 세션**  **CNN 과제** |
|  |
| * 본 과제는 학회 정규 세션 「CNN 1, 2」의 내용을 기반으로 한 **프로젝트 형태의 과제**이며, CNN 개념을 실제로 적용하는 데 초점을 두고 있습니다. 해당 과제는 평가를 위한 것이 아니므로, 학회원 간 토론 및 Slack 질의응답을 적극 활용하여 해결해주십시오. 단, 답안 표절이나 LLM의 남용은 금지합니다. * **CNN\_AssignmentGuide.pdf**을 꼭 먼저 읽고 과제를 진행해주시기 바랍니다. * 코딩 문제들은 주어진 .ipynb 파일들과 .py 파일들에 답안을 작성하시고, 서술형 문제들은 .pdf파일로 작성해 제출해 주십시오. * **2/27 (목) 23시 59분까지 CNN\_AssignmentGuide.pdf**에서의 **제출방법을** 참고하여 Github에 **하나의 .zip 파일로 묶어 제출**해 주십시오. Github에 제출하는 방법을 모른다면 학술부장 혹은 과제 질의응답을 위한 오픈채팅방을 적극적으로 활용해 주십시오. |

**문제 1.**

모델의 학습을 train, validation, test로 나눠서 진행하는 이유는 무엇인가요?

(힌트 : 굳이 validation을 끼워넣는 이유는 무엇인가요?)

만약 validation set이 없다면 test set만을 활용해 모델을 조정하게 되어 test set에 오버피팅될 가능성이 있기 때문에 train set과 test set사이에 validation set을 두어 모델을 조정하고 최종적인 평가는 test set으로 진행해 일반화 성능을 높인다.

**문제 2.**

모델의 generalization을 높여주기 위해 Augmentation을 사용할 수 있다고 했습니다.

PyTorch 구현상으로는 train, validation, test 모든 단계에 Augmention를 구현할 수 있습니다만, 실제로는 특정 단계에서만 Augmention를 수행하게 됩니다. 어느 단계에서 RandAugment가 적용되어야하는지와, 그 이유에 대해서 설명해주세요.

RandAugment는 Train 단계에서만 적용되어야 한다. Validation 또는 Test 단계에서 Augmentation을 적용하면 평가 기준이 달라져 모델 성능을 정확하게 평가할 수 없기 때문이다. 따라서, Augmentation은 학습 단계에서만 적용하고 Validation과 Test 단계에서는 원본 데이터를 그대로 사용해야 모델의 성능을 정확하게 평가할 수 있다.

**문제 3.**

**CustomImageDataset**과 **DataLoader** 클래스의 차이점에 대해서 설명해주세요.

(힌트 : 두 클래스의 리턴 형식에 어떤 차이가 있나요? 어디서부터 Batch 단위로 리턴되나요?)

CustomImageDataset은 개별 데이터를 로드하는 역할을 하며 \_\_getitem\_\_을 호출하면 하나의 이미지와 해당 라벨(단일 샘플)을 반환하여 데이터를 한 개씩 불러오는 기능을 담당한다.

DataLoader는 CustomImageDataset에서 반환하는 개별 데이터를 Batch 단위로 묶어 반환하는 역할을 한다. 예를 들어 batch\_size=32로 설정하면 한 번에 32개의 이미지와 라벨을 가져온다.

CustomImageDataset은 개별 샘플을 리턴하고, DataLoader는 데이터를 배치 크기만큼 묶어서 리턴하여 배치 처리는 DataLoader에서 이루어진다.

**문제 4.**

주어진 과제는 **Binary Classification Task**입니다.

이중 분류를 위해서는 손실함수로 **Binary Cross Entropy**를 사용할 수 있는데, PyTorch에는 Binary Cross Entropy를 학습에 사용할 수 있는 2가지 방법이 있는데요, **'nn.BCELoss'**와 **'nn.BCEWithLogitsLoss'**입니다. 이 둘은 같아 보이지만 구현상에서 명백한 차이점이 있습니다. 이 둘의 차이점에 대해서 서술해주세요.

nn.BCELoss (Binary Cross Entropy Loss)는 시그모이드(sigmoid) 함수가 적용된 출력값(0~1 사이의 확률값)을 입력으로 받는다. 따라서 모델의 출력에 torch.sigmoid()를 적용한 후 이 값을 nn.BCELoss에 전달해야 한다.

nn.BCEWithLogitsLoss (Binary Cross Entropy with Logits Loss)는 시그모이드 함수를 내부적으로 포함하고 있어 모델의 원본 출력(logits)을 바로 입력할 수 있다. 따라서 torch.sigmoid()를 적용하지 않고 모델의 출력값을 그대로 넣어도 된다.

**문제 5.**

**run\_no\_augmentation.ipynb**(또는 **train.py**)에서, 학습이 끝난 이후 val\_loss, train\_loss 변화 추이 그래프와 val\_acc, train\_acc 변화 추이 그래프를 첨부해주세요.(results 폴더 참조)

사용한 hyperparameter들도 명시해주세요.

텍스트, 라인, 도표, 그래프이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

텍스트, 라인, 도표, 그래프이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.**

**문제 6.**

**run\_no\_augmentation.ipynb** (또는 **eval.py**)에서, **best\_model\_no\_augmentation.pth** 를 이용해 test data에서의 confusion matrix와 accuracy를 첨부해주세요. (results 폴더 참조)

텍스트, 스크린샷, 직사각형, 도표이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**문제 7.**

**run\_augmentation.ipynb** (또는 **train.py --augment**)에서, 학습이 끝난 이후 val\_loss, train\_loss 변화 추이 그래프와 val\_acc, train\_acc 변화 추이 그래프를 첨부해주세요.(results 폴더 참조)

사용한 hyperparameter들과 사용한 augmentation(**CustomDataset.py**)도 명시해주세요.

텍스트, 스크린샷, 그래프, 라인이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

텍스트, 라인, 스크린샷, 그래프이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.**

**문제 8.**

**run\_augmentation.ipynb** (또는 **eval.py --augment**)에서, **best\_model\_augmentation.pth** 를 이용해 test data에서의 confusion matrix와 accuracy를 첨부해주세요. (results 폴더 참조)

**test data에서 93% 이상의 accuracy**가 나오도록 해주세요.

텍스트, 스크린샷, 직사각형, 도표이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

**문제 9.**

문제5와 문제7의 그래프들을 통해, **data augmentation**을 적용하지 않았을 때와 적용했을 때의 학습 양상을 비교해 주세요.

Augmentation을 적용하자 Validation Loss의 급격한 변화가 다소 완화되었고, train loss가 첫번째 epoch에서 작게 나타났다. 하지만 Accuracy의 변동성이 여전히 크며 전체적인 양상은 비슷하다.

**\*\***(Optional)

**Data augmentation을 적용한 모델**에 대해, **최고의 성능**이 나오게끔 적절한 data augmentation과 hyperparameter들을 설정해주세요. 그 결과를 첨부해 주세요.

|  |  |
| --- | --- |
| Reference | **Data Science Lab** |
| - 24-1 CNN (10기 윤형진)  - 24-2 CNN (11기 김현진) | 담당자: 12기 이정우  [leejeongwoo9941@yonsei.ac.kr](mailto:leejeongwoo9941@yonsei.ac.kr) |