

인공지능개론 2023 프로젝트 메뉴얼

김연진

2023년 12월 11일

차 례

1	개요	1
1.1	프로젝트의 목적	1
2	데이터셋 설명	1
2.1	클래스 구성	1
2.2	데이터 정보	2
3	프로젝트 일정 및 제한 사항	2
3.1	제출 기한	2
3.2	제한 사항	2
4	제출 형태 및 채점 기준	3
4.1	제출 형태	3
4.2	채점 기준	3

1 개요

1.1 프로젝트의 목적

- 다양한 폰트로 구성된 영문자 및 숫자를 분류하는 문제를 딥러닝을 이용하여 해결한다.
- 프로젝트를 통해 다음과 같은 내용을 학습한다.
 - (1) 딥러닝을 이용한 문제해결을 위한 프로젝트 구조를 이해한다.
 - (2) 목적에 맞는 딥러닝 모델을 구성할 수 있다.
 - (3) 딥러닝 모델을 학습시킬 수 있다.

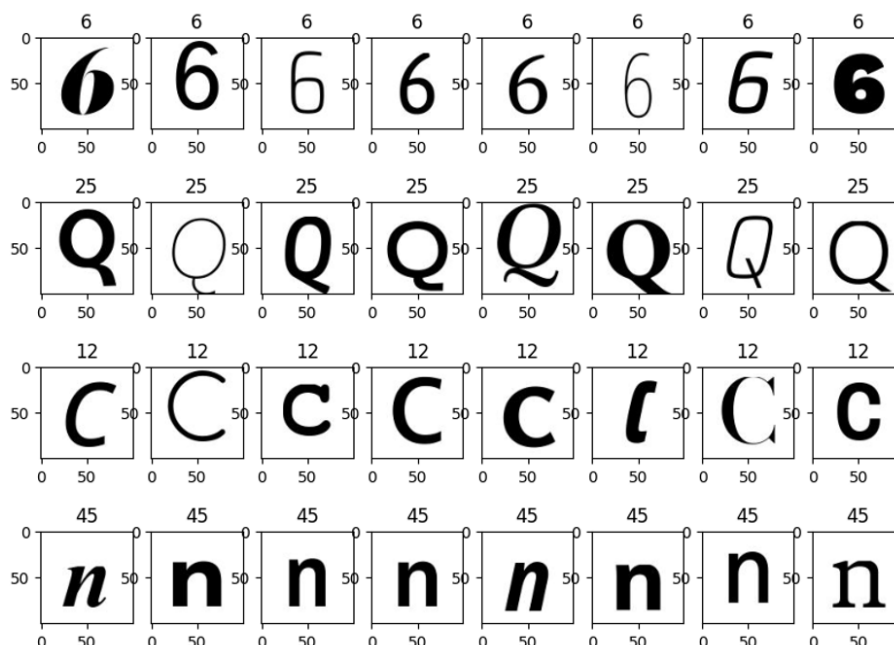


그림 1: 데이터 예시

2 데이터셋 설명

2.1 클래스 구성

- Label 0~9: 숫자(digit)
- Label 10~52: 영문자 (letter)
- 영문자 클래스 중 대/소문자 구별이 어려운 ["c", "k", "l", "O", "p", "s", "v", "w", "x", "z"] class 는 제외했다.

```
{'0': 0, '1': 1, '2': 2, '3': 3, '4': 4, '5': 5, '6': 6, '7': 7, '8': 8, '9': 9,
'A': 10, 'B': 11, 'C': 12, 'D': 13, 'E': 14, 'F': 15, 'G': 16, 'H': 17,
'I': 18, 'J': 19, 'K': 20, 'L': 21, 'M': 22, 'N': 23, 'P': 24, 'Q': 25,
'R': 26, 'S': 27, 'T': 28, 'U': 29, 'V': 30, 'W': 31, 'X': 32, 'Y': 33,
'Z': 34, 'a': 35, 'b': 36, 'd': 37, 'e': 38, 'f': 39, 'g': 40, 'h': 41,
'i': 42, 'j': 43, 'm': 44, 'n': 45, 'o': 46, 'q': 47, 'r': 48, 't': 49, 'u': 50, 'y': 51}
```

그림 2: 데이터 클래스 구성

2.2 데이터 정보

- (1) shape이 (Channels = 1, height = 100, width = 100)인 이미지로 구성되어 있다.
- (2) Train: 41,600, Valid : 15,600 개씩이며, Test 데이터는 채점 시에만 사용하며 공개하지 않는다.
- (3) 각 class별 data 개수는 동일하다. (Totally balanced)
- (4) 데이터셋(Train/Valid) 및 [커스텀 데이터로더 및 유틸](#)을 제공한다.
- (5) [link1](#) / [link2](#)

3 프로젝트 일정 및 제한 사항

3.1 제출 기한

- 제출 마감 : 12월 24일 (일) 자정(23:59)
- 지각 제출 불가

3.2 제한 사항

- 모델의 학습 관련
 - 모델은 Google Colab에서 제공하는 GPU 기준 **15분 이내에 학습 완료**되어야 한다.
 - Train 시간이 15분 이상일 시 초과 시간에 따라 감점한다.
- 모델 관련
 - 모델 파라미터 수는 1,200,000까지로 제한한다.
 - Pretrained model은 사용 불가능하다.
 - 모델 구조는 분류 분야의 논문을 참고해서 구성해도 좋으나, torch 또는 torchvision 등을 통해 불러오지 않고 직접 구현해야 한다.
- 라이브러리 관련
 - Python 표준 라이브러리, Pytorch, Numpy 제공 라이브러리만 모델 구성 및 학습에 사용 가능하다.
 - Matplotlib, Seaborn 등 시각화를 위한 라이브러리는 자유롭게 활용해도 좋다.

- 언급된 것 외의 학습성능 향상을 위한 라이브러리 사용 시 조교에게 문의하라.
- 데이터 관련
 - 데이터 전처리는 자유롭게 진행할 수 있다.
 - 데이터로더는 [프로젝트 도구](#)에서 제공된 것을 가공하지 않고 사용한다.

4 제출 형태 및 채점 기준

4.1 제출 형태

아래의 파일들을 [PJ_학번.zip](#) 으로 압축하여 제출한다. (예: PJ_20181485.zip)

파일명 및 확장자가 잘못될 경우 감점 있음

1. Train에 사용된 주피터 노트북 : [train_학번.ipynb](#)
 - Train 시 할당받은 GPU의 종류를 확인할 수 있도록 [프로젝트 도구](#)에서 제시한 방법대로 `!nvidia-smi` 셀 및 출력을 포함한다.
 - Train의 수행 시간 또한 측정하여 출력한다.
2. Train 후 저장된 model : [학번.pth](#)
3. Test에 사용할 파이썬 스크립트 : [test_학번.py](#)
 train 을 통해 저장한 모델을 불러와서 동작하도록 작성하되, Test set이 주어져 있지 않으므로 Validation set을 test하도록 작성한다.
4. 보고서 : [project_학번.pdf](#)
 - 보고서는 다음 내용을 포함한다.
 1. 과제 목표
 2. 배경 이론
 3. 과제 수행 방법
 4. 결과 및 토의
 할당받은 GPU 종류 및 학습에 사용된 시간을 기재한다. 예를 들어 [Tesla T4 환경에서 2.003초](#)
 모델 파라미터 수를 기재한다.
 5. 참고 문헌

4.2 채점 기준

- Model Accuracy (순위) : 50%
- 프로젝트 보고서 : 50%

```
summary(model, (1, 100, 100))
```

```
=====
Total params: 996,756
Trainable params: 996,756
Non-trainable params: 0
=====
Input size (MB): 0.04
Forward/backward pass size (MB): 7.64
Params size (MB): 3.80
Estimated Total Size (MB): 11.48
=====
```

그림 3: 파라미터 체크 예시