Contribution :

utils.py :

전처리 방법은 <https://wikidocs.net/44249>을 참고하여 구현했습니다. 해당 사이트의 전처리와 다른 방법은 토큰화된 데이터셋의 길이를 데이터셋의 길이 분포를 참고하여 적당한 중간값에서 잘랐으나, 프로젝트에서는 학습 데이터와 테스트 데이터의 토큰화 리뷰 길이의 최댓값이 각각 78, 76인 것을 확인하고 데이터 셋의 길이를 80으로 고정하여 비는 부분을 패딩 하였습니다. 또한 토큰화를 진행하고 룩업 테이블을 매번 생성하는 과정이 상당한 시간을 소요하기 때문에 매 과정을 반복할 필요 없이 json, pkl, npy 등의 확장자로 저장하여 로드하는 방법으로 매번 학습 또는 테스트할 때 시간을 단축할 수 있도록 구현했습니다.

models.py :

Pytorch 프레임워크를 사용해본 경험이 있어서 모델의 경우 Pytorch API Documentaion[https://pytorch.org/docs/stable/index.html]에서 Reference만을 참고하여 구현했습니다.

train.py :

각 모델을 학습할 때 동일한 환경에서 학습을 진행하도록 하기 위하여 train 과정을 진행하는 파이썬 파일을 따로 생성하여 작성했습니다. 또한 학습을 진행할 때 학습 환경을 확인하기 쉽게 하기 위하여 학습 요약 정보를 출력하도록 하였습니다. 매 에폭마다 학습에 소요한 시간, Train Loss 값, Train Acc 값, 검증 데이터 Loss 값, 검증 데이터 Acc 값을 출력하도록 하여 학습 진행 상황을 확인할 수 있도록 하였습니다. 또한 매 에폭당 값들을 저장하여 학습이 끝난 뒤 플로팅을 사용하여 각 값들의 변화 추이를 눈으로 확인하고 쉽도록 시각화하였습니다.