

인공지능 2주차 과제

202055518 김병현

목표

- 모델의 성능 향상

시행 착오

- 처음 주어진 코드를 실행했을 때, **OutOfMemoryError** 발생. 해결 방법

1. 배치 크기 줄이기

배치 크기를 줄이면 한 번에 처리하는 데이터 양이 줄어들어 GPU 메모리 사용을 줄일 수 있다.

2. Gradient Accumulation 활용

배치 크기를 줄이더라도 `gradient_accumulation_steps`를 통해 실질적인 배치 크기를 유지할 수 있다. 이는 여러 미니 배치의 경사도를 축적한 후, 경사도를 업데이트하는 방식이다.

파라미터를 이렇게 변경했다.

```
per_device_train_batch_size=32,  
gradient_accumulation_steps=4,
```

18	1.249600
19	1.159700
20	1.090200

까지 학습을 하다, 사용량 초과로 중단 됐다.
그래서 학습을 빠르게 할 방법을 모색해 봤다.

- 학습을 빠르게 하는 과정

1. 배치 크기 줄이기

배치 크기를 줄이면 한 번에 처리되는 데이터 양이 줄어들어 학습 속도가 빨라질 수 있다.

```
per_device_train_batch_size=16, # 기존 32에서 더 줄이기
```

2. gradient_accumulation_steps 늘리기

배치 크기를 줄였을 때, 학습 성능이 저하될 수 있는데, 성능을 유지하기 위해 `gradient_accumulation_steps`를 늘리면 효과적인 배치 크기를 유지하면서 메모리 사용량을 줄일 수 있다.

```
gradient_accumulation_steps=8, # 기존 4에서 더 늘리기
```

3. num_train_epochs 줄이기

학습을 여러 번 반복할수록 시간이 더 많이 걸린다. 에포크 수를 줄이면 학습 시간이 단축된다.

num_train_epochs=2, # 기존 3에서 2로 줄이기

4. save_steps 조정

너무 자주 모델을 저장하면 학습이 지연될 수 있으므로, save_steps 값을 크게 설정하여 저장 횟수를 줄인다.

위 사항들을 적용해보니, Training Loss 가 1.71 이 나왔다. 학습 성능이 좋지 않은 결과이다.

- 학습 성능을 더 올리기 위한 과정

1. 학습률 (learning_rate) 조정

학습률이 너무 낮거나 높을 경우 학습이 제대로 진행되지 않을 수 있다. 성능이 좋지 않았다면 학습률을 조금 높여서 더 빠른 수렴을 기대할 수 있다.

2. 최적화 알고리즘 변경

만약 더 나은 성능을 원한다면 기본 AdamW(adamw_torch)를 사용할 수 있다.

위 사항을 적용해보니, 학습량이 62로 꽤 많았지만, 학습 속도는 빨랐다.

Training Loss : 1.81

성능을 더 올려야 할 필요가 있다.

```
training_args = TrainingArguments(  
    'nsmc_phi',  
    optim="paged_adamw_32bit",  
    per_device_train_batch_size=32,  
    gradient_accumulation_steps=4,  
    ...  
    num_train_epochs=2,  
    learning_rate=5e-5,  
    ...  
)
```

최적화 알고리즘, 배치 크기, 경사도, 에포크, 학습률을 수정했다.

Training Loss : 1.24

1 아래로 가는 게 목표이기에, 더 진행했다.

```
training_args = TrainingArguments(  
    'nsmc_phi',  
    optim="paged_adamw_32bit",  
    per_device_train_batch_size=16,  
    gradient_accumulation_steps=8,  
    ...  
    save_steps=500,  
    num_train_epochs=2,  
    learning_rate=3e-3,  
    ...  
)
```

배치 크기, 경사도, **save_steps**, 학습률을 수정했다.

Training Loss : 0.0227

학습 평가 : 0.745

느낀점

- 모델 학습을 하는 데 많은 파라미터가 연관되어 있다는 걸 알게 되었다.
- 사용량이 제한된 환경에서 학습 성능을 가져오기 위해, 다양한 방법을 시도할 수 있었다.
- 학습 평가를 더 올리기 위해, 더 다양한 시도를 해 봐야 할 것 같다.