

Parameter-Efficient Fine-Tuning(PEFT)

- 공개된 코드로 파인튜닝을 해본 결과..
- 환각 현상
- 아직 2번 밖에 돌려보지 않아서 파인튜닝을 성공하지는 못한 상태
- Colab에서 하루 할당 GPU 시간이 정해져있음

The screenshot shows a Google Colab notebook interface with the following details:

- Title:** Untitled1.ipynb
- Code Cell 1:** Contains Python code for initializing an SFTTrainer. It specifies various parameters such as model, tokenizer, train_dataset, eval_dataset, and training arguments like batch size, gradient accumulation steps, and learning rate.
- Code Cell 2:** Shows a warning message from the transformers library about the deprecation of the `evaluation_strategy` parameter.
- Code Cell 3:** Displays progress bars for tokenizing a dataset with 41 examples, showing speeds of 27.16, 21.18, and 105.44 examples/s.
- Code Cell 4:** Prints GPU memory statistics. It calculates the start GPU memory, total memory, and maximum memory available on a Tesla T4 GPU.
- Code Cell 5:** Prints the number of trainable parameters, which is 41,943,040.
- Code Cell 6:** Prints training and validation loss values for step 1: Training Loss 1.221700, Validation Loss 6.921641.

제목이 LLM 키우기인 이유

- 양질의 데이터 셋을 공급해야만 잘 대답함. (잘 이해하도록 문서를 정리하여 공급하는 것이 필요)
- RAG, 파인튜닝 모두 코드 베이스를 만들고, 코드를 개선해가는 과정이 필요
- GPU도 필요(Colab 환경 사용)
- 데이터 생산 -> 파인튜닝과 RAG 활용 방법을 꾸준하게 develop 해야 좋은 답변을 기대할 수 있음
- 사내 지식을 활용한 LLM과 코드 리뷰 봇을 만들어서 꾸준하게 develop 해보는것 어떨지 제안

