

Llama2 기반 수학적 추론 성능 비교

김석민, 한민현, 박재홍, 문찬영, 김남수
서울대학교 전기정보공학부 뉴미디어통신공동연구소

{smkim, mhhan, jhpark, cymoon}@hi.snu.ac.kr, nkim@snu.ac.kr

Llama2-based mathematical reasoning performance comparison

Seok Min Kim, Min Hyun Han, Jaehong Park, Chan Young Moon, Nam Soo Kim
Department of Electrical and Computer Engineering and INMC, Seoul National Univ.

요약

본 논문은 llama2 기반 수학적 추론을 실행하고 성능을 비교해본다.

본 논문에서 구현한 알고리즘은 llama2를 기본 모델로 사용하고 hard prompt를 붙여서 사용하거나 다양한 adapter를 붙이고 파인 튜닝하여 성능을 측정한다. GSM8K에 대해 성능을 측정해봤을 때 hard prompt를 사용하는 두가지 방법에 대해 성능은 15.61%와 20.32%를 보여주었다. 또한 다양한 어댑터를 사용하여 성능을 측정해봤을 때 series adapter를 사용한 경우 32.45%, parallel adapter를 사용한 경우 41.02%, LoRA adapter를 사용한 경우 44.66%의 성능을 보여주었다.

I. 서론

인공지능 기술의 발전은 자연어 처리(NLP) 분야에 혁명을 가져왔으며, 이 중 대규모 언어 모델(LLM)의 등장은 특히 주목할 만하다. 이 모델들은 기존의 언어 이해 및 생성 능력을 크게 초월하며, 복잡한 언어적 문제 해결에 있어 새로운 가능성을 열어주고 있다.

LLM은 수십억 개의 파라미터를 사용하여 대규모의 텍스트 코퍼스에서 학습되며, 이를 통해 모델은 고도의 언어 이해와 생성 능력을 갖추게 된다. 이러한 모델은 Transformer 아키텍처를 기반으로 하며, self-attention을 활용하여 문맥을 효과적으로 파악한다.

LLM은 챗봇, 자동 번역, 요약 도구, 검색 엔진 최적화 등 다양한 어플리케이션에서 사용되고 있다. 이를 통해 사용자 경험을 개선하고, 정보 접근성을 높이며, 언어 장벽을 허물어가고 있다.

본 논문에서는 llama2 모델을 사용하여 수학적 추론을 실행하고 성능을 비교해보았다. Hard prompt를 붙여서 사용하거나 다양한 어댑터를 붙이고 파인 튜닝하여 성능을 측정하였다.

II. 본론

2.1 모델

Llama2[1]는 Meta AI에서 개발한 대규모 언어 모델이다. Llama2는 transformer 기반 아키텍처를 사용한다. 본 논문에서는 llama2의 7b 모델을 기본 모델로 사용하였다. 이어지는 장에서는 본 논문에서 사용한 hard prompt와 어댑터에 대해 설명하고자 한다.

2.1.1 Hard prompt

Hard prompt는 LLM과 같은 자연어 처리 모델을 활용할 때 사용되는 특정 유형의 프롬프트이다. 이

용어는 주로 모델의 입력에 대한 구체적이고 명시적인 지침을 포함한 프롬프트를 지칭한다. Hard prompt는 모델이 수행해야 할 작업을 분명하게 설명하고, 종종 구체적인 출력 형식이나 구조를 명시하여, 모델이 특정한 형태의 응답을 생성하도록 유도한다.

Hard prompt의 특징에는 여러가지가 있는데 첫째로는 구체적인 지시어를 포함한다는 점이다. 둘째는 출력 형식을 지정할 수 있으며 마지막으로 특정 작업 수행을 중심으로 구성된다.

2.1.2 Adapter

LLM을 효율적으로 파인 튜닝할 때 사용하는 어댑터는 모델의 사전 학습된 구조와 파라미터를 대부분 그대로 유지하면서, 작은 조정을 통해 특정 작업에 맞게 성능을 최적화하는 방법이다. 어댑터는 모델의 각 레이어 사이에 추가되는 소규모의 신경망으로 구성되며, 이를 통해 새로운 작업에 대한 학습에 필요한 파라미터의 수를 크게 줄일 수 있다. 이 접근 방식은 특히 LLM의 유지 및 배포 비용을 절감하는 데 도움이 된다.

어댑터에는 여러 종류가 있는데 본 논문에서는 어댑터로 series adapter, parallel adapter, LoRA adapter를 사용하였으며 파인튜닝시 llama2의 파라미터는 동결시키고 adapter의 파라미터만 학습을 하였다.

Series adapter는 transformer의 각 레이어 또는 특정 구성 요소(예: self-attention, feed-forward Network) 사이에 순차적으로 삽입되는 어댑터이다. Parallel adapter는 각 레이어 내에 병렬로 추가되는 구조이다. LoRA 어댑터는 low-rank adaptation 기술을 사용하여 transformer 내의 특정 매트릭스(예: attention 또는 feed-forward 매트릭스)에 병렬적으로 삽입되는 어댑터이다.

2.2 실험 및 결과

2.2.1 Hard Prompt

본 논문에서는 [2]의 논문에서 발췌한 두가지 hard prompt 를 사용하였다. Hard prompt 는 질문의 끝에 붙일 수도 있고 답변의 처음에 붙일 수도 있다. 결과는 다음과 같다.

Model	Performance (Acc)
Baseline	14.6
Q_end (Let's work through this problem step-by-step:)	15.61
A_begin (Take a deep breath and work on this problem step-by-step.)	20.32

표 1 Baseline 과 hard prompt 성능 비교(%)

Baseline 의 경우 zero-shot 일 경우 정답을 하나도 맞히지 못하여 8-shot 의 정확도이고 나머지 두개의 결과는 zero-shot 의 성능이다.

2.2.2 Adapter

본 논문에서는 LLM-Adapters[3]에서 구축한 math_10k 데이터셋을 파인튜닝의 트레이닝셋으로 사용하였다. math_10k 데이터셋은 GSM8K, AQuA, MAWPS 의 트레이닝셋으로 구성된 데이터셋이다. 테스트셋으로는 GSM8K, AQuA, MAWPS, SVAMP 의 테스트셋들을 사용하여 각각의 성능을 측정하였다. 결과는 다음과 같다.

Adapter	GSM8K	AQuA	MAWPS	SVAMP
Series	32.45	17.32	77.73	42.3
Parallel	41.02	18.90	83.61	43.8
LoRA	44.66	18.90	82.35	46.8

표 2 Adapter 성능 비교(%)

각 레이어 사이에 순차적으로 삽입되는 series adapter 보다는 병렬적으로 추가되는 구조를 가지는 parallel adapter, LoRA adapter 의 성능이 더 좋은 것으로 나타났다. 이를 통해 어댑터를 병렬적으로 삽입하고 파인튜닝하는 것이 더 좋은 성능을 달성할 수 있다는 것을 알 수 있었다.

III. 결론

본 논문에서는 llama2 모델을 사용하여 수학적 추론을 실행하고 성능을 비교해보았다. 단순히 모델에 입력할 때 hard prompt 를 붙여서 사용하기만해도 성능이 크게 상승하는 것을 확인할 수 있었다. 또한 어댑터의 성능을 비교해봤을 때 병렬적으로 추가되는 구조를 가진 어댑터의 성능이 더 좋은 것을 알 수 있었고 추가적으로 LoRA adapter 의 성능이 가장 좋다는 것을 알 수 있었다.

ACKNOWLEDGMENT

이 논문은 2024 년도 BK21 FOUR 정보기술 미래인재 교육연구단에 의하여 지원되었음.

참 고 문 헌

- [1] Hugo Touvron et al., "Llama 2: Open Foundation and Fine-Tuned Chat Models," in *arXiv:2307.09288*, 2023.
- [2] Chengrun Yang et al., "Large Language Models as Optimizers," in *ICLR 2024*, 2024.
- [3] Zhiqiang Hu et al., "LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models," in *EMNLP 2023*, pp. 5254- 5276, 2023.