RAG End2End 모델에서 LoRA기법을 이용한 성능 향상에 관한 연구

김민창[1], 염세훈[2]
[1]동서울대학교 컴퓨터소프트웨어학과 학부생
[2]동서울대학교 컴퓨터소프트웨어학과 교수
minchang1205@naver.com, shyeom@du.ac.kr

Research on Performance Improvement Using LoRA Techniques in RAG End2End Models

Min-Chang Kim[1], Sae-Hun Yeom[2]
[1]Dept. of Computer Software, Dong-seoul University
[2]Dept. of Computer Software, Dong-seoul University

요 약

본 논문은 RAG(Retrieval-Augmented Generation) End2End의 리소스(Resource) 과부하 문제를 해결하는 동시에 모델 성능을 향상 시키기 위해 PEFT(Parameters-Efficient Fine-Tuning)기술인 LoRA(Low Rank Adaptation)적용에 관한 연구이다. 본 논문에서는 RAG End2End 모델의 파라미터 값과 개수를 유지하면서, LRM(Low Rank Matrices)을 이용하여 추가적인 파라미터만을 미세 조정하는 방식으로, 전반적인 모델의 효율성을 극대화하는 방안을 제시하였다. 본 논문에서 다양한 도메인에 데이터 셋에 대한 제안 방식의 성능을 검증하고자 Conversation, Covid-19, News 데이터 셋을 사용하였다. 실험결과, 훈련에 필요한 파라미터의 크기가 약 6.4억개에서 180만개로 감소하였고, EM(Exact Match)점수가 유사하거나 향상되었다. 이는 LoRA를 통한 접근 법이 RAG End2End 모델의 효율성을 개선할 수 있는 효과적인 전략임을 증명하였다.

1. 서론

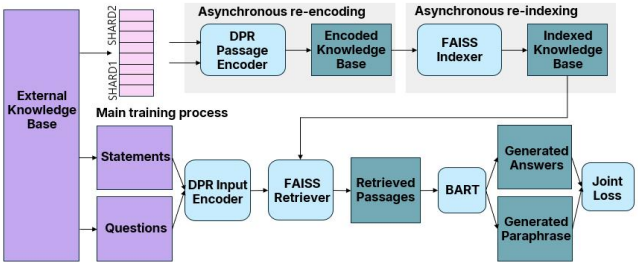
최근 검색 증강 생성 모델인 RAG는 ODQA(Open Domain Question Answering) 태스크에서 주목할 만한 발전을 이루어왔다. 이러한 모델들은 정보의 바다에서 정확한 답을 찾아내는데 탁월한 능력을 보여주며, 특히 Wikipedia와 같은 대규모 지식 베이스를 학습함으로써 그 가능성을 입증해왔다[1]. 그럼에도 불구하고, 기존의 RAG 모델이 Wikipedia에 최적화되어 있어 다양한 전문 분야에 적용 시 한계를 드러내고 있다. 그렇기에 Knowledge Base의 모든 구성 요소를 업데이트 하는 RAG의 확장 버전인 RAG End2End로 연구가 진행되었다[2]. RAG의 생성 부분은 다양한 LLM(Large Language Model)들로 대체 가능하지만 현재 LLM 모델의 파라미터 크기는 지속적으로 증가하는 추세로 인해 리소스 과부하를 야기하고 있다. 이를 해결하고자 다양한 방법들이 연구되어 오고 있고 본 논문에서는 그 방법으로 LoRA기법을 사용하는 방식을 제안하고자 한다.

2. RAG (Retrieval-Augmented Generation) End2End

RAG End2End 모델의 기본적인 구조는 Knowledge Base에 접근하는 검색 부분에 DPR(Dense Passage Retrieval) Question Encoder와 DPR Passage Encoder로 구성되어 있으며 생성 부분은 Bart-Large 모델을 사용한다. RAG End2End가 기존의 RAG와 다른 점은 2가지인데 첫 번째로 미세 조정 단계에서 Passage Encoder의 가중치가 고정 되는 것이 아닌 학습을 통해 지속적으로 업데이트 되어 Knowledge Base를 비동기 적으로 업데이트 한다는 점이다. 두 번째로 보조 학습 신호를 추가하여 QA와 의역 신호의 손실을 최적화 하기 위해 입력 문을 재구성한다는 점이다[2].

2.1 Knowledge Base의 비동기 적인 업데이트

RAG End2End는 특정 도메인에 적용하기 위한 미세 조정 과정이 Passage Encoder의 index를 업데이트하는 방식으로 학습이 진행된다. 이 과정에서, Retriever와 Generator로 학습하는 Main training process를 통해 계산된 DPR Passage Encoder를 사용해 별도의 GPU에서 Knowledge Base를 Re-encoding 한다. 그리고 FAISS (Facebook Ai Similarity Search)를 사용하여 업데이트 된 인코딩으로 Re-indexing 과정을 통해 Knowledge Base를 도메인에 맞게 최적화를 진행한다. 이런 절차는 그림 1과 같다[2].



(그림 1) RAG End2End의 학습 모델 구조

2.1 입력 문의 재구성

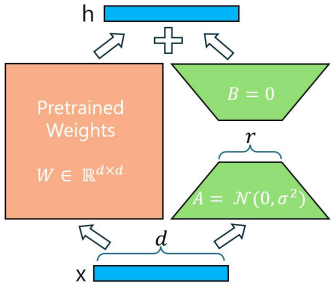
RAG End2End 모델이 다양한 도메인 별 지식을 얻도록 문장을 재구성 시 보조 신호를 추가하게 된다. 이 과정은 먼저 DPR Question Encoder를 사용해 입력에 따른 Encoding을 진행하고 Retriever가 가장 많이 검색한 문장의 유사성 검색을 수행하여 Knowledge Base에서 유사한 문서를 찾게 된다. 이렇게 선별된 문서 세트를 바탕으로 입력 구문을 재구성하게 되며, 이 과정에서 QA 신호와 구별하기 위해 “<p>” 토큰을 추가한다[3]. 그림2는 입력 문의 재구성의 방식에 관한 것이다.

QA INPUT: ⟨Question⟩ + ⟨Retrieved Passages⟩
RECONSTRUCTION INPUT: ⟨<p>⟩ + ⟨Retrieved Passages⟩

(그림 2) 입력 재구성의 방식

3. LoRA을 이용한 미세 조정(Fine-Tuning)

LoRA는 기존의 모델 파라미터를 고정하면서 추가적인 파라미터들을 Low Rank Decomposition Matrices로 학습하는 방법이다. 기존의 입력 크기가 d라고 가정했을 때 $d \times r$ 크기의 Down projection 행렬 A와 $r \times d$ 크기의 Up projection 행렬 B를 추가하여 연산을 수행한다. 이 때 A는 가우시안 분포로, B는 0으로 초기화 한다. 그 후 LoRA는 Backward 시에 Low Rank의 행렬로 가중치를 근사하여 계산하게 된다[4].



(그림 3) LoRA 미세조정(Fine-Tuning) 과정

4. 제안한 RAG End2End와 LoRA의 혼합 방식

기존 RAG End2End 모델은 검색 부분의 2개의 Encoder와 생성 부분의 Bart-Large 모델 모두 Transformer 구조의 주요 특징인 어텐션 메커니즘(Attention Mechanism)을 가지고 가중치가 계산이 된다. 본 논문에서는 제안한 LoRA 기법을 활용하기 위해 어텐션 메커니즘에서 중요한 역할을 하는 Query, Key, Value에 각각 Low Rank의 행렬 A, B를 추가하였다. 수식 (1)(2)(3)은 Query, Key, Value의 처음 가중치 값인 상태로 고정 시킨 후 A, B로 가중치를 계산하는 식이다.

$$W'_Q = W_Q + A_Q B_Q^T \quad \cdots\cdots\cdots\cdots (1)$$

$$W'_K = W_K + A_K B_K^T \quad \cdots\cdots\cdots\cdots (2)$$

$$W'_V = W_V + A_V B_V^T \quad \cdots\cdots\cdots\cdots (3)$$

이러한 방식은 추가 행렬 A, B만으로 가중치를 계산하기 때문에 실제 훈련에 필요한 파라미터 수가 줄어들어 사용되는 GPU 메모리가 감소하면서도 성능을 유지하거나 향상 시킬 수 있게 되었다.

5 제안 방식의 성능 평가를 위한 실험

제안 방식의 성능 평가를 위해 사용된 각 도메인 별 데이터 셋에 대한 설명은 표 1과 같다.

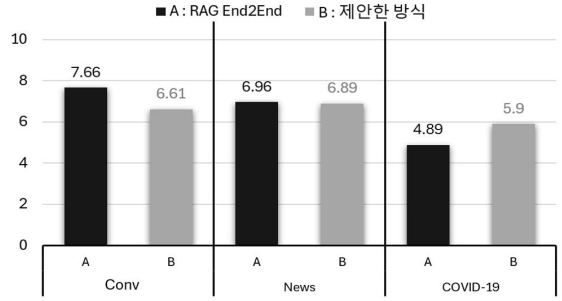
<표 1> 데이터 셋 도메인에 대한 설명

데이터 셋 별 도메인	내 용
대화 Q/A	QAConv 데이터 셋에서 제공된 10,000개의 대화를 각각 최대 100개의 단어로 구성된 구절로 분할하여 110,000개의 구절로 구성된 외부 지식 베이스를 생성
뉴스 Q/A	NewsQA 데이터 셋에서 제공된 10,000개의 뉴스 기사를 사용하여 85,000개의 100단어 구절을 지식 베이스로 생성
COVID-19 Q/A	CORD-19에서 제공된 5000개의 과학 논문 전문을 바탕으로 250,000개의 100단어 구절을 지식 베이스로 생성

실험을 수행한 결과인 표 2에서는 훈련 가능한 파라미터 수가 줄어 GPU 메모리 사용량이 줄어드는 것을 볼 수 있는데 이는 COVID-19의 데이터 셋을 사용했을 때의 비교이기 때문에 데이터 셋에 따라 약간의 변동이 있을 수 있다. 훈련 가능한 파라미터 수가 약 6.4억개에서 약 180만개로 감소하였고 메모리 사용량이 22GB에서 5GB로 감소하였다. GPU 사용량 감소로 인해 기존의 RAG End2End 모델은 Colab의 A100으로 훈련해야 했지만 본 논문에서 제시한 LoRA를 활용한 방식은 T4만으로 훈련이 가능하였다.

<표 2> 데이터 셋 도메인에 대한 설명

모델 명	훈련 가능한 파라미터 수	GPU 메모리 사용량
RAG End2End	약 6.4억개	약 22GB
제안한 방식	약 180만개	약 5GB



(그림 4) 데이터 셋에 따른 EM Score

그림 4는 제안 방식의 EM Score가 Conv 데이터 셋에서는 감소하지만 News와 COVID-19 데이터 셋에서는 유사하거나 오히려 증가함을 알 수 있다. 훈련 가능한 파라미터 수와 GPU 메모리 사용량 감소에 비해 성능 감소 부분은 미비한 것으로 생각된다.

6. 결론

본 논문에서 제안한 방식은 다양한 전문 분야에 맞게 적응하도록 구성한 RAG End2End의 생성 부분이 LLM의 지속적인 크기 증가에 따라 적응할 때 발생하는 리소스 과부하 문제를 해결하면서도 성능 저하를 방지하는데 중점을 두었다. 이러한 접근은 On Device AI의 ODQA 태스크에서 활용할 수 있을 것으로 예상된다. 이는 사용자의 직접적인 상호작용을 필요로 하는 애플리케이션에서 AI 모델의 효율성을 극대화 할 수 있을 것으로 생각된다. 향후 연구는 본 논문에서 제안한 방식을 Llama2, Gemini, GPT-4 모델에 적용하여 성능 향상을 목표로 진행하고자 한다.

참고 문헌

[1] Tom Kwiatkowski., "Natural questions: a benchmark for question answering research", Transactions of the Association for Computational Linguistics 7:453 - 466, 2019

[2] Shamane Siriwardhana., "Improving the Domain Adaptation of Retrieval Augmented Generation(RAG) Models for Open Domain Question Answering", arXiv:2210.02627, 2022

[3] Nitish Shirish Keskar., "Ctrl: A conditional transformer lan-guage model for controllable generation", arXiv:1909.05858, 2019

[4] Edward J.Hu., "LoRA: Low-Rank Adaptation of Large Launguage Models", arXiv:2106.09685, 2021