음성인식 기반의 맞춤형 로컬 챗봇(ChatBot) 시스템 구현에 관한 연구

김성진[1], 임채우[1], 염세훈[2]
[1]동서울대학교 컴퓨터소프트웨어학과 학부생
[3]동서울대학교 컴퓨터소프트웨어학과 교수
ho1582@naver.com, dlacodn456@naver.com, shyeom@du.ac.kr

Research on implementing customized local ChatBot system based on speech recognition

Sung-jin Kim[1], Chae-woo Im[1], Sae-Hun Yeom[1]
[1]Dept. of Computer Software, Dong-seoul University

요 약

본 논문에서는 오픈소스(Open Source) LLM(Large Language Model)인 Llama3을 이용하여 음성으로 동작하는 맞춤형 로컬 챗봇을 구현하고자 한다 이를 위해 PEFT(Parameter Efficient Fine-Tuning) 방식과 RAG(Retrival Augmented Generation) 방식을 혼합하는 하이브리드 방식을 사용해 Llama3을 파인 튜닝하고, Ollama을 이용하여 로컬 컴퓨터에서 동작하는 챗봇을 구현하였다. 챗봇의 배포를 위해 서버 부분은 LangServe와 Ngrok을 사용하였고 클라이언트 부분은 모바일 환경에서 원활히 동작하는지 확인하기 위해 교육용 임베디드 시스템인 Raspberry Pi 5를 사용하여 구현하였다. 또한 사용자의 편의성을 위해 음성인식 기능을 추가하였다. 또한 구현한 챗봇에 성능평가를 진행하였다. 성능 측정 방식은 정확도를 사용하였고, 데이터 셋은 총 18개로 각 쿼리마다 5번씩 총 90개의 쿼리로 성능 평가를 진행하였다. 성능 평가 결과 하이브리드 방식이 가장 우수한 성능을 보여주었다.

1. 서론

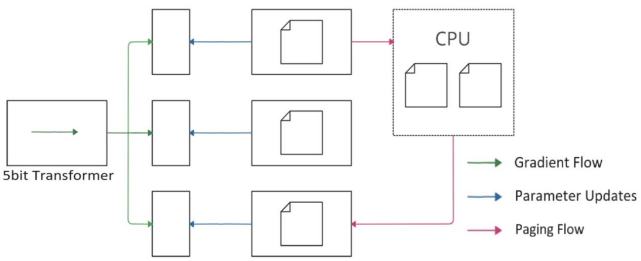
공공 기관이나 기업에서 맞춤형 챗봇을 구현할 때, 기존에는 폐쇄형 LLM(ChatGPT, Gemini, Claude 등)의 API를 활용한 방식이 일반적이었다. 그러나 이 방식은 기업정보 보안 및 API 사용료 부담 등의 문제점을 가지고 있다. 또한 트랜스포터 디코더 모델의 환각현상 또한 성능 저해하는 요소로 작용하였다. 이러한 문제를 해결하기 위해 본 논문에서는 메타(Meta)사의 Llama3 모델을 활용하여 해결하였고, 환각현상은 RAG와 PEFT를 결합한 하이브리드 접근법으로 특정 도메인에 맞는 맞춤형 챗봇을 구현하였다. 나아가, 구현한 맞춤형 챗봇이 모바일 환경에서 원활히 동작하는지 확인하기 위해 Raspberry Pi 5를 활용하여 구현하였다. 사용자의 편의성을 높이기 위해 음성인식 기능을 추가하였다[1].

2. 개방형 LLM인 Llama3

Llama3은 Meta 사에서 개발한 Llama 모델의 3번째 버전으로 오픈소스, 프리 라이선스로 제공되는 대규모 언어 모델이다. Llama3은 비교적 적은 매개변수로 빠른 출력의 생성이 가능한 8B 모델과 복잡합 작업에 능숙한 70B 모델로 구분된다. Llama3을 개발할 때 사용된 주요 기법들에는 GQA(Group-Query-Attention)과 RLHF(Reinforcement Learning Human-Feedback), Qutization이 있다[2].

3. PEFT(Parameter Efficient Fine-Tuning)

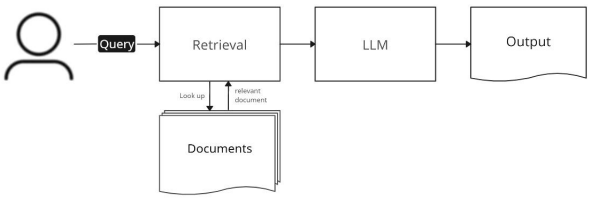
대규모 언언 모델의 크기가 작을 때는 전체 파라미터를 사용해 전이학습하는 Full Fine-Tuning 방식이 가능하였다. 그러나 모델의 성능이 향상되면서 파라미터수가 증가했고, 이로 인해 Full Fine-Tuning의 비용과 시간이 증가하면서 실제 구현이 어려워졌다. 이러한 문제를 해결하기 위해 PEFT가 등장했다. PEFT는 대부분의 파라미터를 동결시키고, 적은 양의 파라미터만을 사용해 효과적으로 학습하는 전이학습 방식이다. 본 논문에서는 PEFT 방식 중 하나인 QLoRA(Quantized Low-Rank Adaptation)를 사용하였다. QLoRA의 동작도는 그림 1과 같다[3].



(그림 1) QLoRA방식의 동작 블록도

4. RAG(Retrival Augmented Generation)

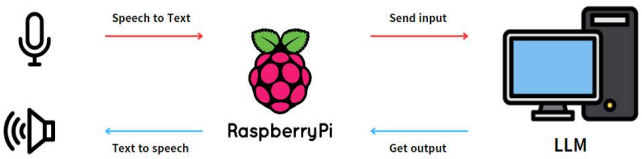
LLM은 기본적으로 환각현상 문제를 가지고 있다. 이를 해결하기 위해 많은 비용과 시간을 필요로 하는 파인튜닝 방식 대신에 보다 적은 비용과 시간으로 환각현상을 해결하는 방식인 RAG라는 시스템이 등장했다. 본 논문에서는 별다른 최적화를 거치지 않은 Live RAG를 사용하였다. RAG의 동작 블록도는 그림 2와 같다.



(그림 2) RAG방식의 동작 블록도

5. 맞춤형 로컬 챗봇 시스템의 전체 구성도

본 논문에서 구현한 맞춤형 로컬 챗봇 시스템은 클라이언트와 서버 두 부분으로 구성되어 있다. 클라이언트 부분은 음성인식 및 합성 기능을 기반으로, 모바일 환경에서 원활히 동작하도록 설계되었다. 이를 위해 교육용 임베비드 보드인 Raspberry Pi 5를 사용하여 구현하였다. 서버 부분은 Llama3 8B 모델을 로컬 컴퓨터에서 구동시키기 위해 Ollama, LangServe와 Ngrok을 활용하여 구축하였다. 맞춤형 로컬 시스템의 구성도는 그림 3과 같다. 사용한 Raspberry Pi 5의 구현환경은 표1과 같다.



(그림 3) 맞춤형 로컬 챗봇시스템의 구성도

<표 1>. 임베디드 시스템(Raspberry PI 5)의 환경

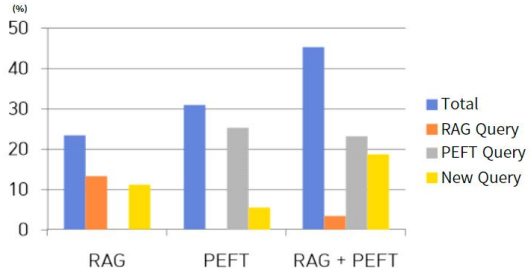
항목		내용
H/W	CPU	BCM2712 (2.4GHz)
	GPU	VideoCore VII (800MHz)
	MEMORY	SDRAM 4267
	SD card	micro 카드 슬롯, SDR104 고속 모드 지원
S/W	O/S	Debian GNU/Linux 12
	Library	langchain_community = 0.2.7 speechrecognition = 3.10.7 langserve = 0.2.2, pyttsx3 = 2.90

6. 맞춤형 로컬 챗봇 시스템의 서버모듈의 성능평가

맞춤형 로컬 챗봇 시스템의 성능 평가 실험은 동서울대학교 컴퓨터소프트웨어학과의 안내 챗봇 시스템 개발을 목표로하였다. LLM은 Llama3의 8B을 사용하여 RAG, PEFT, 두 방식을 하이브리드 방식에 대해 성능 평가를 진행하였다. 평가 지표는 정확도를 사용하였고, 사용한 데이터셋은 RAG 시스템에 사용한 PDF에서 6개, PEFT 학습시 사용한 데이터 셋에서 6개, 새로운 쿼리 6개를 사용하였다. 각 쿼리마다 5번씩 총 90번 질문을 진행하였다. 성능평가 결과는 표 2와 같고 각 방식별 성능평가 결과 그래프는 그림 4와 같다.

<표 2>. 챗봇서버 모듈의 각 방식별 성능평가결과

측정 \ 방식	RAG	PEFT	RAG+PEFT
Accuracy	23.3%	30.8%	45.1%



(그림 4). 각 방식별 성능평가 결과 그래프

맞춤형 로컬 챗봇 시스템의 각 방식별 성능평가 결과를 보면 RAG 방식이 23.3%로 가장 낮은 성능을 보여주었다. 다음으로 PEFT 방식은 30.8%를 RAG 방식과 PEFT을 혼합한 하이브리드 방식은 45.1%의 결과를 얻었다. 성능평가를 통해 알 수 있는 내용은 RAG와 PEFT를 동시에 사용하는 것이 더 좋은 성능을 낸다는 것을 알 수 있었다.

7. 결론

본 논문에서는 오픈소스 LLM인 Llama3 8B을 이용하여 음성으로 동작하는 맞춤형 로컬 챗봇을 구현하였다. 챗봇은 클라이언트 부분과 서버 두 부분으로 구현하였다. 클라이언트 부분은 음성인식 및 합성 기능을 기반으로, 모바일 환경에서 원활히 동작하도록 설계하였다. 서버 부분은 모델이 로컬 컴퓨터에서 동작하게 하기 위해 LangServe와 Ngrok을 사용하여 구현하였다. 이렇게 구현한 시스템의 성능을 평가하기 위해서 RAG, PEFT, 하이브리드 접근법 3가지에 대해 정확도를 사용하여 비교 평가를 진행하였다 사용한 데이터 셋은 18개로 각 쿼리당 5번씩 총 90개의 쿼리로 진행하였다. 성능 평가 결과 3가지 방법중에서 하이브리드 접근법이 가장 우수한 성능을 보여주었다. 본 논문에서 구현한 시스템은 추후 기관이나 기업에서 맞춤형 로컬 챗봇을 구현하고자 할 때 활용하면 좋을 것으로 기대한다. 또한 실험 결과에서 RAG에 대한 성능이 좋지 못하였다. 이유는 본 논문에서 사용한 RAG 시스템은 최적화가 되어 있지 않는 Naive RAG을 사용하였기 때문이라 생각하여 이후 연구에 AutoRAG 프레임워크를 활용하여 RAG 시스템을 최적화 할 예정이다.

참고문헌

[1] https://bitnine.tistory.com/586

[2] Portakal, E. (2024, May 10). Llama 3 by Meta Ai 리뷰. TextCortex. https://textcortex.com/ko/post/llama-3-review

[3] Lingling Xu et al, “Parameter-Efficient Fine-Tuning Methods for Pretrained Language Models: A Critical Review and Assessment” arXiv:2312.12148, 2023