B2G 제안요청서(RFP) 도메인 특화 QA를 위한 RAG와 SFT의 실용성 비교 연구

Comparative Analysis of RAG and SFT for Domain-Specific QA: A Case Study on B2G Request for Proposals

김진욱 (Jinuk Kim)
Codeit AI Engineer Bootcamp

Abstract

본 연구는 높은 정확도와 신뢰성이 요구되는 B2G(Business-to-Government) 공공입찰 및 제안요청서(RFP) 도메인에서, 도메인 특화 질의응답(QA) 시스템 구축을 위한 두 가지 대표 접근인 **검색 증강 생성(RAG)**과 **지도 미세 조정(SFT)**의 **운영 관점 실용성**을 비교한다. 약 100건 내외의 RFP 원천 문서를 수집·전처리하여 벡터DB를 구축하고, 동일한 질문 세트에 대해 **(1) Base LLM 기반 RAG 시스템**과 **(2) 검색 없는 SFT 단독 시스템**의 성능을 측정하였다. **LLM-as-a-Judge 기반 정량 평가(0~5점)** 결과, RAG가 SFT 대비 **정답성(Correctness)**과 **근거 충실성(Faithfulness)**에서 각각 **+0.65, +0.25**우세하였다. 반면 **지연시간(Latency)**은 SFT가 RAG 대비 약**5.9배** 빠르게 측정되어, 품질–속도 간 뚜렷한 상충 관계가 관찰되었다. 추가적으로, 생성 모델을 고정한 상태에서 **후보 확장(k_fetch) + Cross-Encoder 리랭킹 + 최종 top-k 재선택(k_final) + 컨텍스트 길이 제한(max_chars)**을 적용한 검색 파이프라인 보조 실험을 수행하였다. 그 결과 평균 점수 상승이 관찰되었으나, 일부 문항에서는 **검색된 근거가 답변에 충분히 반영되지 않는 현상**도 확인되었다. 본 결과는 RFP 도메인에서 **정확성과 근거 기반성**이 중요한 경우 RAG가 기본 선택이 될 가능성이 높음을 보이되, **실시간성이 중요한 환경에서는 SFT가 대안**이 될 수 있음을 시사한다. 향후 과제로는 근거 인용(또는 발췌) 강제, 컨텍스트 기반 자동 평가의 엄격화, 표본 확장 등을 제안한다.

1. 서론 (Introduction)

1.1 연구 배경

공공입찰 및 제안요청서(RFP) 문서는 복잡한 규정·요건·예외 조항을 포함하며, 수치·기간·조건 등 **정밀 정보가 다층적으로 서술**된다. 이 도메인에서 QA 시스템의 작은 오류는 단순한 정보 전달 실패를 넘어 **법규 위반, 입찰 탈락** 등 실무 리스크로 이어질 수 있다. 따라서 본 과업에서는 유창성(Fluency)보다 **근거 기반 정확성(Accuracy)**과 **신뢰성**이 핵심 성능 축이 된다.

1.2 연구 목적

도메인 지식을 LLM에 주입하는 대표 방법은 **(i) 외부 지식을 검색**해 답변에 반영하는 **RAG**와 **(ii) 도메인 데이터로 모델 파라미터를 업데이트**하는 **SFT**이다. 본 연구는 실무 관점에서 아래 연구 질문을 검증한다.

1. **Primary RQ :** 동일한 B2G 입찰 도메인 환경에서, **RAG 시스템이 SFT 단독 시스템보다** 정확성/근거 충실성/스타일/응답 속도 측면에서 **실용적으로 우수한가?**
2. **Secondary RQ :** 검색 파이프라인에 **후보 확장 및 Cross-Encoder 리랭킹**을 추가하면, 기본 RAG 대비 **유의미한 성능 향상**을 제공하는가?

2. 시스템 및 실험 환경 (System & Experimental Setup)

2.1 데이터셋 및 인덱싱 (Corpus & Indexing)

약 100건 내외의 실제 B2G 입찰 공고 및 RFP 문서(hwp/pdf)를 수집하였다. 전처리(텍스트 추출) 및 청킹(Chunking)을 통해 총 **8,516개 청크**를 생성하고, Chroma 기반 벡터DB를 구축하였다. 임베딩 모델은 OpenAI `text-embedding-3-small`을 사용하였다.

2.2 SFT 데이터 구축 및 학습 방법론 (SFT Data Construction & Training Methodology)

SFT 모델의 효과적인 **도메인 적응(Domain Adaptation)**을 위해 다음과 같은 파이프라인을 구축하였다.

첫째, **데이터셋 구축(Data Engineering)** 단계에서는 **전처리된 RFP 문서 청크를 GPT-4에 입력**하여, 입찰 컨설턴트 관점의 '**질문(Instruction)**'과 그에 대한 '**답변(Output) 쌍**'을 생성, 약 500~1,000건의 **Instruction Tuning Dataset**을 구축하였다.

둘째, **베이스 모델(Base Model)**로는 한국어 입찰 용어 이해도가 우수한 `beomi/Llama-3-Open-Ko-8B`를 선정하였다.

셋째, **학습(Training)** 단계에서는 컴퓨팅 자원의 효율성을 극대화하기 위해 **QLoRA (Quantized Low-Rank Adaptation)** 기법을 적용하였다. Google Colab (T4 GPU) 환경에서 모델을 **4-bit로 로드**한 후, 전체 파라미터의 **약 1~2%에 해당하는 LoRA 어댑터(Adapter)만을 학습**시켜 도메인 지식과 어조(Tone & Manner)를 주입하였다.

마지막으로, 학습된 어댑터를 베이스 모델과 병합(Merge)하고 **llama.cpp**를 활용해 **GGUF 포맷(Q4_K_M)**으로 **양자화 변환**함으로써, 로컬 환경에서도 추론 가능한 경량화 모델을 최종 구축하였다.

2.3 비교 시스템 정의 (Experimental Groups)

본 연구에서 "모델"은 단순 가중치가 아니라, 지식 주입 경로를 포함한 **시스템 구성(System Configuration)**으로 정의한다. 주 실험 비교군은 다음과 같다.

- **RAG-Base (System A)** : Base LLM(Llama-3 한국어, GGUF)에 **벡터 검색(Vector Retrieval)**을 결합한 구성. "**학습 없이 검색만으로 충분한가?**"를 검증한다.
- **SFT-Only (System B)** : RFP 도메인으로 미세 조정된 SFT LLM(GGUF) 단독 구성. 검색 없이 **모델 내부 지식(Parametric knowledge)**만으로 답변한다. **"검색 없이 학습만으로 충분한가?"**를 검증한다.

(주 : SFT+RAG Hybrid 구성은 예비 실험 단계에서 출력 불안정성이 관찰되어 본 비교에서 제외하였다.)

2.4 실행 환경 및 재현성 요약 (Reproducibility)

시스템 비교의 재현성을 위해 추론 및 측정 범위를 간단히 요약한다([표 2]). (세부 파라미터는 실험 노트북/로그에 기록)

[표 2] 실행 환경 및 추론 설정 요약

구분	RAG-Base (System A)	SFT-Only (System B)
추론 엔진	llama.cpp / llama-cpp-python	llama.cpp / llama-cpp-python
모델 형식	GGUF	GGUF
검색 모듈	Chroma Vector DB (top-k=3)	없음
임베딩	text-embedding-3-small	해당 없음
Latency 측정	End-to-End (요청→응답 완료)	End-to-End (요청→응답 완료)

3. 평가 방법론 (Evaluation Methodology)

3.1 평가 데이터셋 구축

벡터DB에 포함된 문서 내용을 바탕으로 총 **20개 In-domain 질의**를 생성하였다. 각 질의에 대해 실험자가 원문을 확인하여 1~3문장 분량의 **레퍼런스 답안(Reference Answer)**을 작성하였다. 본 레퍼런스는 "절대적 진리값"이 아니라 **문서 근거를 사람이 요약한 기준값**임을 전제한다.

3.2 평가 프로토콜 (LLM-as-a-Judge)

평가의 공정성과 효율성을 위해 **LLM(GPT-4o)을 심판(Judge)으로 활용**하는 정량 평가를 수행하였다. 각 항목은 **0~5점 척도**로 평가되었다.

- **정답성 (Correctness)** : 답변이 레퍼런스와 의미적으로 얼마나 일치하는가?
- **근거 충실성 (Faithfulness)** : "근거에 의해 지지되는 주장만 수행했는가?"를 평가한다.
  - **RAG-Base**는 retrieved context를 기준으로, 답변의 핵심 주장(수치/기간/조건)이 컨텍스트에 의해 직접 지지되는지 본다.
  - **SFT-Only**는 외부 컨텍스트가 없으므로, 레퍼런스와의 모순 및 **근거 없이 구체 수치·기간·조건을 단정하는 경향(unsupported specificity)**을 중심으로 평가한다.
  - 따라서 Faithfulness는 두 시스템 간 "완전 등가 비교"라기보다, **근거 기반 QA 관점에서의 위험 신호를 비교하는 지표**로 해석한다.
- **스타일 (Style)** : 입찰 컨설턴트 톤앤매너와 구조를 준수하는가?
- **지연 시간 (Latency)** : 요청부터 응답 완료까지 End-to-End 소요 시간(초)

4. 실험 결과 및 분석 (Results & Analysis)

4.1 주 실험 결과 : RAG-Base vs SFT-Only

20개 문항에 대한 평균 결과는 [표 1]과 같다.

[표 1] 주 실험 평균 성능(0~5점, Latency=초)

시스템 (System)	Correctness	Faithfulness	Style	Latency (s)
RAG-Base	**3.05**	**3.90**	**3.55**	23.20
SFT-Only	2.40	3.65	3.20	**3.95**

- **정답성/충실성 :** RAG-Base가 SFT-Only 대비 Correctness **+0.65**, Faithfulness **+0.25** 높았다. 이는 SFT-Only가 전반적으로 "그럴듯한 설명"을 생성하더라도, **수치·기간·조건(정밀 슬롯)**에서 불일치가 누적되면 Correctness 격차로 이어질 수 있음을 시사한다. 다만 본 해석은 **N=20**의 제한된 표본에서 관찰된 경향이며, 불일치 유형의 체계적 분류는 확장 실험에서 보강한다.
- **스타일 :** Style 차이는 크지 않았다(3.55 vs 3.20). 이는 (i) RAG 시스템 프롬프트만으로도 전문 톤을 상당 수준 유도했거나, (ii) SFT 데이터가 "형식/톤 최적화"보다 "내용 전달"에 더 치우쳤을 가능성을 시사한다.
- **지연시간 :** SFT-Only는 평균 **3.95초**로, RAG-Base(23.20초) 대비 약 **5.9배** 빠르게 측정되었다.

4.2 보조 실험 결과 : 검색/선별 파이프라인 변경의 효과(생성 모델 고정 ablation)

4.2.1 보조 실험의 목적과 해석 범위

본 절은 “검색/선별 단계 개선이 답변 품질을 얼마나 끌어올리는지”를 분리 관찰하기 위한 **ablation**이다. 이를 위해 **생성 모델을 gpt-4o-mini로 고정**하고, **retrieval 파이프라인만 변경**하였다. 또한 보조 실험의 Judge는 overall **0~10점** 스케일을 사용하므로, **4.1(0~5점)과 절대값을 직접 비교하지 않고**, **동일 절 내 상대 비교**로만 해석한다.

4.2.2 검색 파이프라인 구성 비교(기술 명세)

보조 실험에서 비교한 파이프라인은 (A) 기본 벡터검색 기반과 (A′) 후보 확장 및 Cross-Encoder 리랭킹 기반이다.

[표 3] 보조 실험 검색/선별 파이프라인 명세

구분	기본 검색 파이프라인 (A)	후보 확장+리랭킹 파이프라인 (A′)
후보 수집	VectorDB similarity top-k (예: k=3)	VectorDB에서 **더 큰 후보 풀** 우선 수집 (**k_fetch**)
후보 재정렬	없음	**Cross-Encoder Re-ranker**로 (질문, 문서) 관련도 재평가 후 정렬
최종 컨텍스트	상위 k개 그대로 사용	리랭킹 결과 상위 **k_final**만 사용
길이 제어	(없거나 제한적)	**max_chars**로 컨텍스트 길이 제한
생성 모델	**gpt-4o-mini 고정**	**gpt-4o-mini 고정**
비교 의도	“top-k=3만으로 충분한가?”	“더 많이 가져오고 더 잘 고르면 좋아지나?”

> ***용어 정리***
> *- **k_fetch** = “일단 많이 가져오는 후보 수(Recall 확보용)”*
> *- **k_final** = “최종 컨텍스트로 넣는 상위 문서 수(Precision/비용 균형)”*
> *- **max_chars** = “컨텍스트가 너무 길어지는 것 방지(지연/비용/프롬프트 한도 대응)”*

4.2.3 보조 실험 정량 결과(요약)

생성 모델을 고정한 상태에서, 후보 확장+리랭킹 파이프라인(A′)은 평균 점수(0~10)가 **7.06(A)**에서 **8.36(A′)**으로 상승하였다. 이는 특히 기본 top-k에서

근거를 놓치는 **검색 실패(Recall 실패)** 상황에서 A′가 더 자주 근거를 회수했을 가능성과 일관된다.

4.2.4 케이스 분석 : "찾았는데도 답에 못 쓰는" 현상

보조 실험 로그에서 관찰된 핵심 이슈는 두 가지로 요약된다.
1. **리랭킹이 근거 회수에는 도움**이 되지만,
2. **회수된 근거가 답변에 반영되지 않는** 경우가 존재한다("찾아놓고도 안 씀")
이를 대표 케이스로 정리하면 [표 4]와 같다.

[표 4] 케이스 분석(대표 문항) — 검색 성공/실패와 생성 반영 이슈

문항 ID	관찰 요약	해석(원인 후보)	실무적 액션(간단)
14	A는 근거를 놓쳤으나, A′는 리랭킹으로 관련 문서를 회수	**리랭킹이 Recall 개선**에 기여	A′에 "근거 1~2문장 발췌/인용"을 **프롬프트로 강제**
16	A/A′ 모두 레퍼런스 핵심 근거를 회수하지 못함	리랭킹 문제가 아니라 **후보 풀 자체에 정답 근거가 없음**(query/인덱싱/청킹/임베딩 이슈 가능)	질의 재작성(멀티쿼리)·청킹 개선·후보 수(k_fetch) 확대 등 **상류 개선**
19	A′가 그럴듯한 답을 했으나, 컨텍스트 근거는 약함	Judge가 "그럴듯함"에 점수를 주는 **평가 편향 가능성**	Judge를 **컨텍스트 포함 채점**으로 강화 + "근거 없으면 감점" 규칙 명시

5. 고찰 (Discussion)

5.1 정확성과 속도의 상충 관계(Trade-off)

본 결과는 "RAG가 항상 우수하다"라기보다, **목적에 따른 선택 기준**을 제공한다.

- **정확성 우선(리스크 높은 업무):** 보증금, 제출 요건, 기간 등 오류 허용도가 낮은 항목이 핵심인 RFP 분석에서는 RAG-Base가 더 적합하다.
- **속도/비용 우선(응답 지연이 치명적) :** 실시간 응대가 중요하거나 높은 QPS가 요구되는 환경에서는 SFT-Only가 운영 효율상 장점이 있다. 다만 정확성 요구가 높은 질의에는 별도 검증(룰/후처리/근거 확인)이 필요할 수 있다.

5.2 SFT-Only의 한계와 활용 가능성

SFT-Only는 검색 없이도 평균 Correctness 2.4 수준으로, 일정 수준의 도메인 QA가 가능함을 보였다. 그러나 외부 근거가 주어지지 않는 설정에서는 레

퍼런스와 불일치하는 **구체 수치·기간·조건을 단정**하는 응답(unsupported specificity)이 일부 관찰될 수 있으며, 이는 근거 기반 QA에서 운영 리스크로 이어질 수 있다. 따라서 SFT-Only는 (i) **톤/형식 고정**, (ii) **초안 작성**, (iii) **근거 확인이 뒤따르는 보조 시나리오**에 우선 적용하고, 정확성이 요구되는 질의에는 RAG-Base를 기본 구성으로 두는 전략이 합리적이다.

6. 한계 및 타당성 위협 (Threats to Validity)

위협 요인	설명	완화/향후 계획
표본 수 제한	N=20의 In-domain 질의로 일반화 한계	질의 수 확대 및 유형별 층화(stratification)
레퍼런스 요약 편향	레퍼런스 답안이 '원문 요약'이므로 표현/포함 요소에 편차 가능	핵심 슬롯 체크리스트(수치/기간/예외조건) 도입, 근거 발췌문 함께 저장
Judge 편향	그럴듯한 답변을 과대평가할 가능성	컨텍스트 기반 채점 강화, 근거 인용 강제(citation-required) 평가
Faithfulness 등가성	RAG는 컨텍스트 기반, SFT는 컨텍스트 부재로 동일 의미 비교가 어려움	향후 oracle evidence(근거 발췌문)를 모든 시스템에 동일 제공해 등가 비교

7. 결론 및 향후 연구 (Conclusion)

본 연구는 공공입찰 RFP 도메인에서 RAG-Base와 SFT-Only의 실용성을 정량 비교하였다. 주 실험(N=20)에서 RAG-Base는 **정답성/근거 충실성**에서 우위를 보였고, SFT-Only는 **응답 지연시간**에서 큰 강점을 보였다. 보조 실험에서는 후보 확장 및 Cross-Encoder 리랭킹을 포함한 파이프라인이 평균 점수를 개선했으나, 일부 문항에서 **회수된 근거가 답변에 충분히 반영되지 않는 현상** 및 **평가 편향 가능성**이 확인되었다.

향후 연구에서는 (i) 표본 확대 및 질의 유형의 층화, (ii) 답변에 근거 문장 발췌/인용을 강제하는 프롬프트/후처리, (iii) 컨텍스트 기반 자동 평가의 엄격화

(근거 없으면 감점, 정직한 보류 처리 등)를 통해, 근거 기반 QA 시스템의 신뢰성과 재현성을 강화할 필요가 있다.