Adaptive-RAG: Learning to Adapt Retrieval-Augmented Large Language Models through Questions Complexity

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, Jong C.Park

KAIST

NAACL 2024

발표자: 송선영

2024/07/08

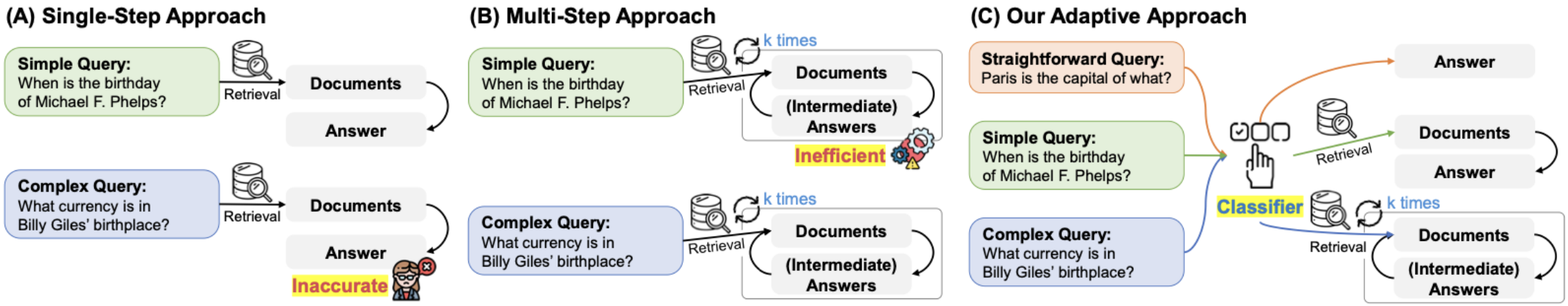
Introduction

- LLM은 Parametric한 knowledge에만 의존하기 때문에 사실과 다른 답을 생성하는 경우가 많음
- 이를 해결하기 위해 검색으로 non-parametric knowledge 를 통합하는 RAG 방법이 제안됨
  - Single-hop QA approach
  - Multi-hop QA approach
- 복잡한 query를 효과적으로 처리하기 위해서는 Multi-step QA 방식이 필요하지만, 많은 비용이 듦
- 또한, Multi-step QA 방식은 복잡한 query는 필수적이지만, 단순한 query에는 불필요한 비용이 발생함

→ 따라서 이 논문에서는 query의 복잡도에 따라 접근 방법을 선택하는 Adaptive QA system인 Adaptive-RAG를 제안함

Adaptive-RAG

- Adaptive-RAG는 **query의 복잡도에 따라 접근 전략을 분류**
  1. No Retrieval (label: A)
     - 외부 문서 참조 x, LLM 자체에서 답변
  2. Single-step Approach (label: B)
     - 한 번의 검색을 통해 외부 정보 활용
  3. Multi-step Approach (label: C)
     - 여러 번의 검색을 통해 여러 정보를 종합해 활용



Adaptive-RAG

- Adaptive-RAG는 query의 복잡도에 따라 전략을 분류하는 **classifier**를 사용
  - 모델: T5-large

  - 학습 데이터셋 구축
    - Labeled dataset이 존재하지 않아 automatic labeling 방법 활용
      - 먼저, 6개의 QA benchmark dataset에서 400개의 query를 샘플링해 3가지 전략으로 분류
        - 예를 들어, non-retrieval 방법으로 정답을 생성했다면 label은 A로 labeling
        - single-step 방법으로 정답을 생성했다면 label은 B로 labeling
    - 만약, 모델이 적절한 답을 생성하지 못해 labeling을 할 수 없다면, benchmark label이 single-step인 경우에는 B, multi-step인 경우에는 C로 labeling

Experimental Setups

- **Datasets**
  - Single-hop QA
    - SQuAD v1.1, Natural Questions, TriviaQA
  - Multi-hop QA
    - MusiQue, HotpotQA, 2WikiMultiHopQA

- **Retriever**: BM25 (Wikipedia)
- **Generator**: FLAN-T5 series models(3B, 11B), GPT-3.5 model (gpt-3.5-turbo-instruct)

- **Metrics**
  - Effectiveness와 Efficiency의 모두 평가
    - For effectiveness
      - F1, EM, accuracy
    - For efficiency
      - Retrieval-Generate step의 수, 각 query에 대한 평균 답변 시간
      - (one-step approach와 비교)

Evaluation – Main results

- Real-world에서는 다양한 수준의 복잡성을 가진 query들이 존재하기 때문에 Adaptive 전략이 필요함
- Adaptive type model들 중 Adaptive-RAG가 가장 좋은 성능을 보임

		FLAN-T5-XL (3B)					FLAN-T5-XXL (11B)					GPT-3.5 (Turbo)				
Types	**Methods**	EM	F1	Acc	Step	Time	EM	F1	Acc	Step	Time	EM	F1	Acc	Step	Time
Simple	**No Retrieval**	14.87	21.12	15.97	0.00	0.11	17.83	25.14	19.33	0.00	0.08	35.77	48.56	44.27	0.00	0.71
	Single-step Approach	34.83	44.31	38.87	1.00	1.00	37.87	47.63	41.90	1.00	1.00	34.73	46.99	45.27	1.00	1.00
Adaptive	**Adaptive Retrieval**	23.87	32.24	26.73	0.50	0.56	26.93	35.67	29.73	0.50	0.54	35.90	48.20	45.30	0.50	0.86
	Self-RAG*	9.90	20.79	31.57	0.72	0.43	10.87	22.98	34.13	0.74	0.23	10.87	22.98	34.13	0.74	1.50
	Adaptive-RAG (Ours)	**37.17**	**46.94**	**42.10**	**2.17**	**3.60**	**38.90**	**48.62**	**43.77**	**1.35**	**2.00**	**37.97**	**50.91**	**48.97**	**1.03**	**1.46**
Complex	**Multi-step Approach**	39.00	48.85	43.70	4.69	8.81	40.13	50.09	45.20	2.13	3.80	38.13	50.87	49.70	2.81	3.33
Oracle	**Adaptive-RAG w/ Oracle**	45.00	56.28	49.90	1.28	2.11	47.17	58.60	52.20	0.84	1.10	47.70	62.80	58.57	0.50	1.03

Evaluation – Main results

- Real-world에서는 다양한 수준의 복잡성을 가진 query들이 존재하기 때문에 Adaptive 전략이 필요함
- Adaptive type model들 중 Adaptive-RAG가 가장 좋은 성능을 보임
- LLM 모델의 종류와 상관없이 Adaptive-RAG는 Multi-step Approach보다 Time을 훨씬 감소시킴

		FLAN-T5-XL (3B)					FLAN-T5-XXL (11B)					GPT-3.5 (Turbo)				
Types	**Methods**	EM	F1	Acc	Step	Time	EM	F1	Acc	Step	Time	EM	F1	Acc	Step	Time
Simple	**No Retrieval**	14.87	21.12	15.97	0.00	0.11	17.83	25.14	19.33	0.00	0.08	35.77	48.56	44.27	0.00	0.71
	Single-step Approach	34.83	44.31	38.87	1.00	1.00	37.87	47.63	41.90	1.00	1.00	34.73	46.99	45.27	1.00	1.00
Adaptive	**Adaptive Retrieval**	23.87	32.24	26.73	0.50	0.56	26.93	35.67	29.73	0.50	0.54	35.90	48.20	45.30	0.50	0.86
	Self-RAG*	9.90	20.79	31.57	0.72	0.43	10.87	22.98	34.13	0.74	0.23	10.87	22.98	34.13	0.74	1.50
	Adaptive-RAG (Ours)	**37.17**	**46.94**	**42.10**	**2.17**	**3.60**	**38.90**	**48.62**	**43.77**	**1.35**	**2.00**	**37.97**	**50.91**	**48.97**	**1.03**	**1.46**
Complex	**Multi-step Approach**	39.00	48.85	43.70	4.69	8.81	40.13	50.09	45.20	2.13	3.80	38.13	50.87	49.70	2.81	3.33
Oracle	**Adaptive-RAG w/ Oracle**	45.00	56.28	49.90	1.28	2.11	47.17	58.60	52.20	0.84	1.10	47.70	62.80	58.57	0.50	1.03

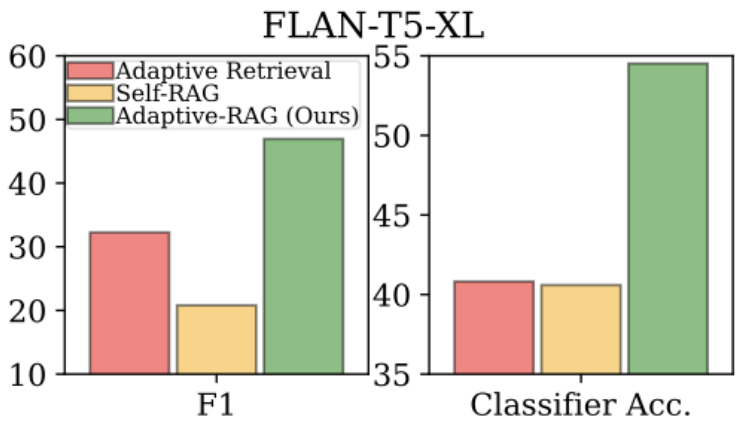
Evaluation – Main results

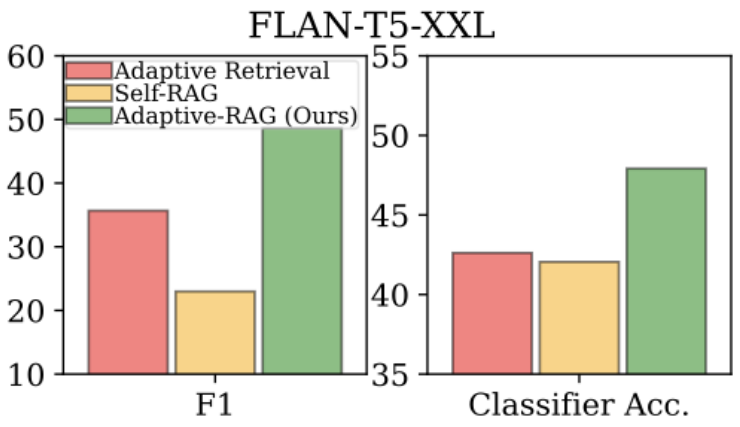
- Real-world에서는 다양한 수준의 복잡성을 가진 query들이 존재하기 때문에 Adaptive 전략이 필요함
- Adaptive type model들 중 Adaptive-RAG가 가장 좋은 성능을 보임
- LLM 모델의 종류와 상관없이 Adaptive-RAG는 Multi-step Approach보다 Time을 훨씬 감소시킴
- Oracle을 사용하게 되면 성능과 시간 효율성 모두 향상될 수 있음
  - Oracle: classification performance is perfect

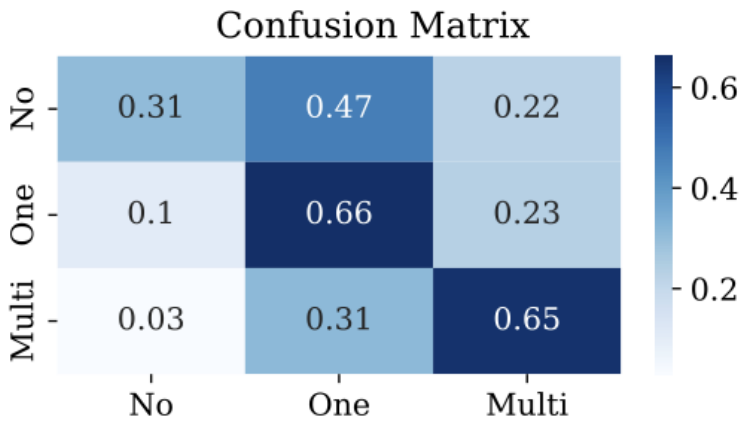
		FLAN-T5-XL (3B)					FLAN-T5-XXL (11B)					GPT-3.5 (Turbo)				
Types	**Methods**	EM	F1	Acc	Step	Time	EM	F1	Acc	Step	Time	EM	F1	Acc	Step	Time
Simple	**No Retrieval**	14.87	21.12	15.97	0.00	0.11	17.83	25.14	19.33	0.00	0.08	35.77	48.56	44.27	0.00	0.71
	Single-step Approach	34.83	44.31	38.87	1.00	1.00	37.87	47.63	41.90	1.00	1.00	34.73	46.99	45.27	1.00	1.00
Adaptive	**Adaptive Retrieval**	23.87	32.24	26.73	0.50	0.56	26.93	35.67	29.73	0.50	0.54	35.90	48.20	45.30	0.50	0.86
	Self-RAG*	9.90	20.79	31.57	0.72	0.43	10.87	22.98	34.13	0.74	0.23	10.87	22.98	34.13	0.74	1.50
	Adaptive-RAG (Ours)	**37.17**	**46.94**	**42.10**	**2.17**	**3.60**	**38.90**	**48.62**	**43.77**	**1.35**	**2.00**	**37.97**	**50.91**	**48.97**	**1.03**	**1.46**
Complex	**Multi-step Approach**	39.00	48.85	43.70	4.69	8.81	40.13	50.09	45.20	2.13	3.80	38.13	50.87	49.70	2.81	3.33
Oracle	**Adaptive-RAG w/ Oracle**	45.00	56.28	49.90	1.28	2.11	47.17	58.60	52.20	0.84	1.10	47.70	62.80	58.57	0.50	1.03

Evaluation – classifier performance

- Adaptive-RAG가 다른 Adaptive method보다 더 높은 분류 정확도를 보임
- 하지만 여전히 잘못된 분류가 있기 때문에 이를 개선하는 것이 Future work



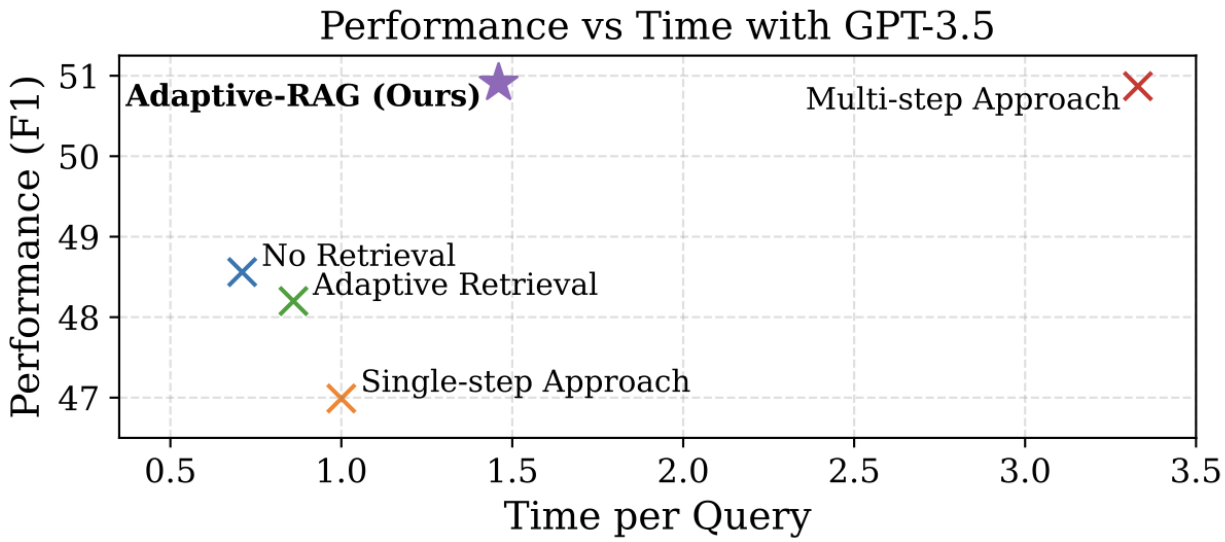




Evaluation

- Classifier 모델의 크기를 줄여도 큰 성능 차이를 보이지 않음
- Adaptive-RAG는 각 query에 대한 평균 시간은 가장 적으면서 가장 높은 성능을 보임으로써, 효율성과 정확도를 모두 향상시킴

	QA		Classifier (Accuracy)			
Sizes	**F1**	**Step**	**All**	**No**	**One**	**Multi**
Small (60M)	45.83	964	53.48	26.65	70.62	53.18
Base (223M)	45.97	983	53.41	26.42	69.46	56.82
Large (770M)	**46.94**	**1084**	**54.52**	30.52	66.28	65.45



Conclusion

- Real-world에서는 다양한 복잡도를 가진 query가 존재
- 본 논문에서는 다양한 복잡도를 가지는 query를 효과적이고 효율적으로 처리하기 위해 Adaptive-RAG를 제안
- Adaptive-RAG 는 query의 복잡도를 기반으로 적절한 접근 전략을 dynamic하게 분류하여 정확성과 효율성을 향상시켰음

- Limitations
  1. Classifier를 학습하기 위한 데이터셋을 구축하기 위해 automatic labeling 방법을 사용하는데, 이는 잘못된 label을 만들 수 있음
     - 따라서 향후 연구에서는 classifier의 정확도를 개선하고, Question-Answer 쌍의 label 외에도 더 다양한 범위의 복잡도 데이터셋 구축을 목표로 함

  2. Real-world에서는 사용자의 입력이 불쾌하거나 유해한 경우가 있는데, 이 때 불쾌감을 주는 문서가 검색되고 이에 의해 부적절한 응답이 생성될 수 있다는 문제점이 존재함
     - 이를 해결하기 위해, RAG 프레임워크 내에서 사용자의 입력과 검색 문서 모두에서 부적절한 콘텐츠를 감지하고 관리하는 방법을 연구하는 것이 필수적

Thank You

감사합니다.