The Distracting Effect: Understanding Irrelevant Passages in RAG

Chen Amiraz[1*], **Florin Cuconasu**[1,2†‡], **Simone Filice**[1§], **Zohar Karnin**[1¶]

[1]Technology Innovation Institute, [2]Sapienza University of Rome

ACL 2025

Main Problem

검색된 콘텐츠는 때때로 문제가 되는 행동을 유발

- 검색이 항상 성공적인 것은 아니며, 이러한 경우 중 다수 입력문서에는 관련없는 구절이 포함

⟹ 관련 없지만 의미론적으로 관련된 정보를 포함하여 LLM을 오도하고 따라서 답변 생성을 해칠 수 있음

쿼리에 대해 LLM에 대한 구절의 산만 효과를 어떻게 평가?

- passage 자체의 distracting 효과를 분리

⟹ 측정된 distracting 효과가 높을수록 골드 구절과 함께 입력에 포함될 때 RAG 정확도가 더 많이 감소

- **쿼리별로 가장 방해가되는 문서들을 모아, 학습 => RAG robustness 개선**

Distracting Passages

Measuring the Distracting Effect

- LLM에게 구절을 기반으로 q에 답변하도록 요청, passage에 q에 대한 답변이 포함되지 않은 경우 "NO-RESPONSE"를 출력하도록 함

$$\mathrm{DE}_q(p) = 1 - p^{\mathrm{LLM}}(\text{NO-RESPONSE}|q, p)$$

- DE_q(p) 를 동일한 쿼리와 관련된 구절의 상대적 순위 점수로 해석

Obtaining Distracting Passages

Retrieving Distracting Passages (E5-Base)

- Standard Retrieval (R_{st})

  : 검색 시스템에서 상위에 랭크되었으나 실제로는 정답과 무관한 구절을 선택

- Answer-Skewed Retrieval (R_{sk})

  : 질문과는 관련이 있지만 정답과는 무관한 문서를 찾기 위해 쿼리 임베딩을 변형

  정답 임베딩 E_D(a) 를 빼거나 투영(Projection)하는 방식을 사용

  - 쿼리와 관련 있지만 답변과는 관련 없는 문서를 검색하는 아이디어를 표현하는 산술적 방법

$$E^{\text{sub}}(q,a) = E_Q(q) - \lambda E_D(a)$$

$$E^{\text{proj}}(q,a) = E_Q(q) - \lambda \frac{\langle E_Q(q), E_D(a) \rangle E_D(a)}{\|E_D(a)\|^2}$$

Obtaining Distracting Passages

Generating Distracting Passages

- 타연구에서 사용되는 다양한 유형의 distractor 범주를 사용해 **범주별로 모델의 취약점을 공략하는 'Hard Negatives'를 인위적으로 제조**

1. Related Topic (G_{rel})

: 질문의 주제와 밀접하지만 답은 포함하지 않는 내용 (예: 인물의 아들에 대한 설명)

2. Hypothetical (G_{hypo})

: 가상의 상황이나 과거의 상황을 가정하여 다른 답을 유도하는 내용

3. Negation (G_{neg})

: "흔히 하는 오해와 달리 ~가 아니다"와 같이 오답을 부정문 형태로 제시 (ex: 학생들이 소득에 대해 세금을 납부하지 않는다는 것은 흔한 오해입니다)

4. Modal Statement (G_{modal})

: "~일 가능성이 있다"와 같이 불확실성을 전제로 오답을 제시 (ex: 피라미드는 벽돌, 흙, 모래로 이루어진 경사진 둘러싸는 둑을 이용하여 건설되었을 수 있습니다)

- Claude로 Passage 생성 후, NLI 모델로 생성된 passage가 쿼리와 entailment 관계인 샘플은 필터링

Analyzing the Distracting Effect

Distracting Effect of Retrieved Passages

- 일반 ODQA에 대해 Wikipedia psg100m으로 검색
- **순위와의 상관관계**
  : 검색 순위가 높을수록(Top-rank) 해당 무관 구절의 방해 효과가 더 강력
- **리랭커(Reranker)의 역설 (+ 표기: 검색 후 reranking한 경우)**
  : 리랭킹 단계를 추가하여 검색 성능을 높이면, 역설적으로 살아남은 무관 구절들은 LLM을 더 잘 속이는 '더 강력한 방해꾼 ' 이 됨

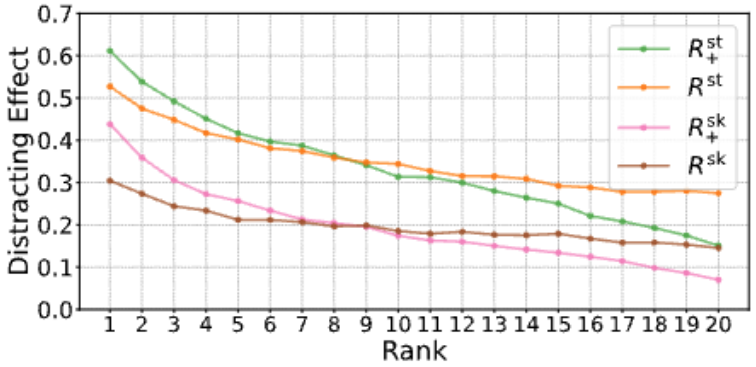


Figure 2: Average distracting effect at different rank positions for various retrieval methods. Results are shown for Llama-3.1-8B, averaged across datasets. Higher-ranked passages consistently demonstrate greater potential to mislead the model. Similar trends were observed across all tested LLMs (see Figures 20 and 21).

우리가 성능이 좋다고 믿고 사용하는 최신 검색기와 리랭커는, 관련 문서를 잘 찾는 만큼이나 LLM을 가장 완벽하게 속일 수 있는 '최상급 방해꾼'들을 상위 순위에 배치하는 부작용을 가짐

Analyzing the Distracting Effect

Comparing Distracting Effects of Different Methodologies

- 비교 방식:
  I각 질문당 검색에서 찾은 가장 강력한 구절 1개와, 4가지 범주(G_{rel}, G_{hypo}, G_{neg}, G_{modal})별로 생성된 구절 1개씩을 가져와 방해 효과(DE) 점수 비교

- **표준 검색 + 리랭커 (R_{st}+)가 평균적으로 가장 높은 방해 효과**
- 생성기법 중에서는 G_{modal} (확실치 않은 진술)이 가장 강력
- 모델간의 높은 상관관계

⟹Llama로 측정한 방해 점수가 Qwen이나 Falcon에서도 거의 비슷하게 높음

검색 기반이 평균적으로 강하긴 하지만,
질문의 절반 가까이는 생성된 구절(가설, 부정문 등)이 더 효과적으로 모델을 속임

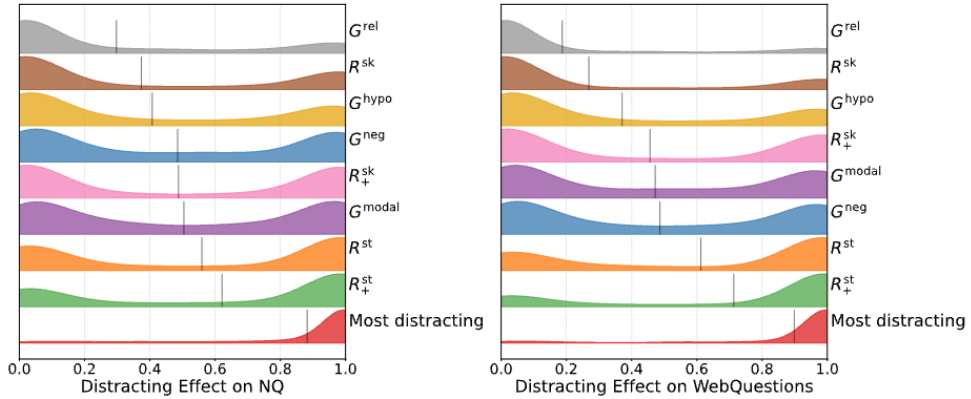
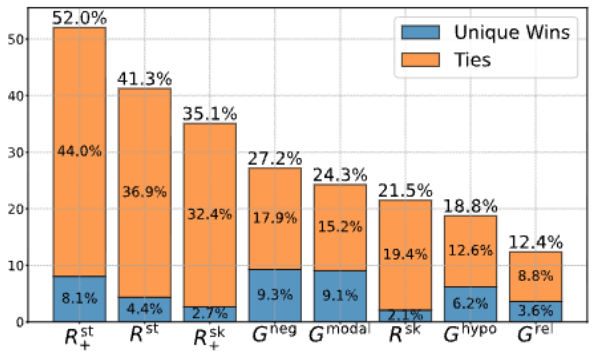


Figure 1: Distribution of distracting effect for passages obtained through different methods, as measured by Llama-3.1-8B. Methods are ordered by their mean distracting effect (shown by vertical black lines), with higher means indicating a greater ability to distract the model.



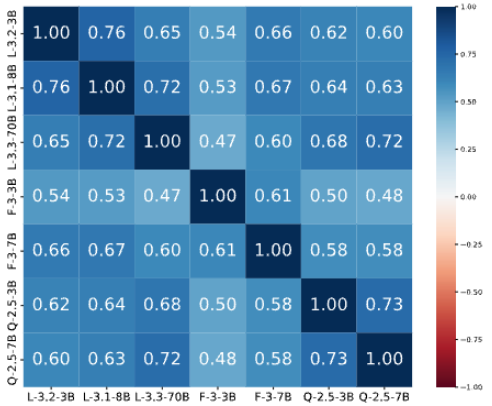


Figure 4: Spearman correlation of distracting effect computed using different LLMs (abbreviated, e.g., Llama → L). The strong correlations suggest that the distracting effect of a passage is relatively consistent across models despite architectural differences.

Prompting with Relevant and Distracting Passages

관련 구절과 방해 구절의 혼합

- Hard Distracting (HD): 방해 효과 점수가 0.8 이상인 구절 (매우 강력한 방해꾼)
- Weak Distracting (WD): 방해 효과 점수가 0.2 이하인 구절 (별로 안 위협적인 구절)

⟹Gold 문서와 함께 주입하여 성능 변화 측정

- **정확도의 급격한 하락:**
  정답 문서만 줬을 때보다, Hard Distracting(HD) 구절을 하나 섞었을 뿐인데 답변 정확도가 6~11%p 하락
- **약한 방해꾼과의 차이:**
  Weak Distracting(WD) 구절은 정확도에 거의 영향을 주지 않거나 하락 폭이 미미.
  정의한 DE 점수가 실제 모델의 성능 저하를 예측하는 매우 신뢰할 수 있는 지표임을 뒷받침
- **모델 규모와의 관계:**
  흥미롭게도 Llama-3.3-70B와 같은 거대 모델조차도 Hard Distracting 구절이 옆에 있으면 정답 문서를 앞에 두고도 오답을 내뱉는 취약성

LLM	Only Gold	Gold + WD	Gold + HD
Llama-3.2-3B	82.6	79.4	71.5
Llama-3.1-8B	80.6	80.1	73.9
Llama-3.3-70B	81.1	80.1	75.2
Falcon-3-3B	78.5	74.1	67.1
Falcon-3-7B	84.1	81.5	73.3
Qwen-2.5-3B	80.9	75.5	69.4
Qwen-2.5-7B	82.4	80.4	73.7

Table 1: Answer accuracy when prompting the LLM with the gold passage only, gold passage with a weak distracting passage (WD), and gold passage with a hard distracting passage (HD). Values that are NOT underlined are different in a statistically significant way w.r.t. the gold-only case (Wilcoxon test with p-value < 0.01).

Application: RAG Fine-Tuning

Fine-Tuning 후 RAG 성능비교

- **Retrieve**: 일반적인 검색 시스템이 가져온 Top-5 문서를 학습에 사용
- **Rerank**: 리랭커까지 거친 정교한 Top-5 문서를 학습에 사용
- **Hard** (제안 방법): 가장 강력한 방해 구절들을 학습 데이터에 삽입
  질문 중 50%는 [진짜 문서 1개 + 강력한 방해 구절 4개]로 구성
  나머지 50%는 아예 정답 없이 [강력한 방해 구절 5개]로만 구성하여 모델이 낚이지 않는 법을 배우게 함

- **Key Takeaway 1:**
  가짜 정보가 섞여 있어도 진짜 정보를 골라내는 능력이 비약적으로 상승
- **Key Takeaway 2:**
  특히 검색 결과에 정답이 없을 때, 가짜 정보에 낚여 헛소리를 하는 대신 자신의 지식을 정확히 활용하는 능력이 좋아짐(Hallucination 감소)

Test Set	Train Set	Llama-3.2-3B			Llama-3.1-8B		
		acc_u	acc_g	acc	acc_u	acc_g	acc
NQ	*None*	15.2	51.4	37.9	12.8	56.7	40.3
	Retrieve	13.3	**57.1**	40.7	21.0	62.4	46.9
	Rerank	11.5	56.5	39.7	19.9	**63.2**	47.0
	Hard	**21.4**	55.6	**42.8**	**32.0**	59.8	**49.4**
PopQA	*None*	8.4	55.7	35.9	8.7	61.3	39.3
	Retrieve	8.8	62.6	40.1	16.6	73.2	49.5
	Rerank	9.6	60.3	39.1	18.0	**75.1**	**51.2**
	Hard	**14.9**	**63.6**	**43.2**	**21.6**	71.2	50.4
TriviaQA	*None*	38.0	79.2	67.8	39.8	86.4	73.5
	Retrieve	36.9	79.4	67.6	56.8	87.1	78.7
	Rerank	33.3	76.7	64.7	58.4	**87.5**	79.4
	Hard	**54.1**	**82.3**	**74.5**	**68.9**	87.0	**82.0**
WebQA	*None*	21.4	54.4	41.9	19.0	53.8	40.6
	Retrieve	20.7	55.1	42.1	28.4	**59.9**	48.0
	Rerank	20.3	54.1	41.3	30.4	59.6	48.6
	Hard	**35.0**	**58.7**	**49.7**	**36.8**	59.7	**51.0**

Table 2: Answer accuracy averaged over all 4 test sets. *None* is the non-fine-tuned baseline, *Retrieve*, *Rerank* and *Hard* are fine-tuning strategies. Metrics: (1) acc_u, accuracy on ungrounded instances, (2) acc_g, accuracy on grounded instances, and (3) acc, overall accuracy. Bold values indicate the highest per model and dataset. The LLMs fine-tuned on the *Hard* dataset achieve statistically significant superior acc in all test sets besides Llama-3.1-8B on PopQA where results are slightly lower than training on *Rerank*, but in a non-statistically significant manner (Wilcoxon test with p-value < 0.01).

Less Is More: Elevating RAG via Performance-Driven Context Compression

Ziqiang Cui [1] **Yunpeng Weng** [2] **Xing Tang** [3] **Peiyang Liu** [4] **Shiwei Li** [2] **Bowei He** [5] **Jiamin Chen** [1]
Yansen Zhang [1] **Xiuqiang He** [3] **Rui Zhang** [2] **Chen Ma** [1]

Motivation

긴 컨텍스트로 인해 발생하는 높은 계산 비용과 LLM의 'lost-in-the-middle' 현상과 같은 문제에 직면

- 문서 수를 0개에서 10개로 늘리면 인코딩 토큰이 0개에서 1428개로 급증

기존 압축 방식은 휴리스틱에 의존하여 성능 저하를 초래

- 소스 텍스트와 요약본 간의 상호 정보량을 최대화하거나, 교사 모델의 출력을 모방하거나, BM25와 같은 어휘 중첩 지표를 기반으로 콘텐츠를 선택하는 방식
- 일반적인 요약에는 효과적이지만, 이러한 학습 목표는 궁극적인 성능을 고려하지 못함
  => 이러한 결함은 성능을 위한 최적의 요약이 무엇인지 나타내는 ground-truth signal이 없기 때문에 발생 => **ground truth signal 주어서 RL하자**

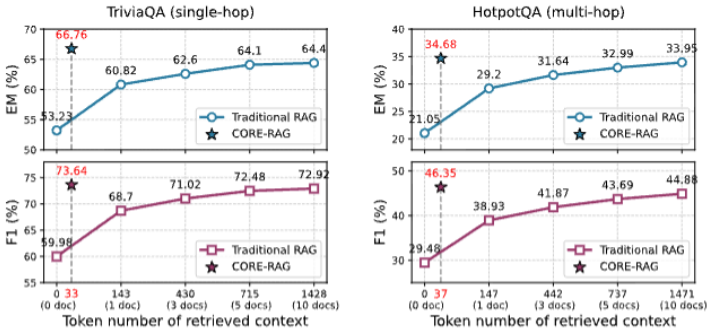


Figure 1. Performance evolution with an increasing number of retrieved documents on two datasets. Traditional RAG requires more documents for better performance, while our method CORE-RAG achieves superior results with significant context compression.

CORE-RAG

문제정의

- 입력 질문 q, 타겟 출력 y, 그리고 k개의 검색된 문서 집합 D가 주어졌을 때,
  목표는 q와 관련하여 D를 가장 유용한 정보를 보존하면서 훨씬 적은 토큰을 사용하는 요약 s로 압축하는 것

- 압축기가 policy model. 즉, 요약을 생성하는 compression model
- Reader LLM이 압축기의 요약을 입력으로 받아서 정답 생성을 수행

Pipeline

Distillation

- DeepSeek-V3을 교사 모델로 활용하여 학습 데이터셋의 각 질문과 관련된 검색된 문서 D의 요약 s을 생성
- 이후 Q+s 의 RAG > Q의 RAG 인 샘플만 선정
- 해당 샘플들을 활용하여 IT Model을 SFT. 이때 모델은 Top-5 passage를 한번에 보고 이에 대한 요약을 생성
- Qwen2.5-1B-Instruct

Reinforcement Learning

- GRPO (Reader LLM은 고정. Policy 모델만 학습)
- Reader LLM은 Q + s 를 입력받아서 answer를 생성하고, 이를 기반으로 EM + F1 reward를 산출
- Qwen2.5-14B-Instruct / Llama-3.1-8B-Instruct

Overview

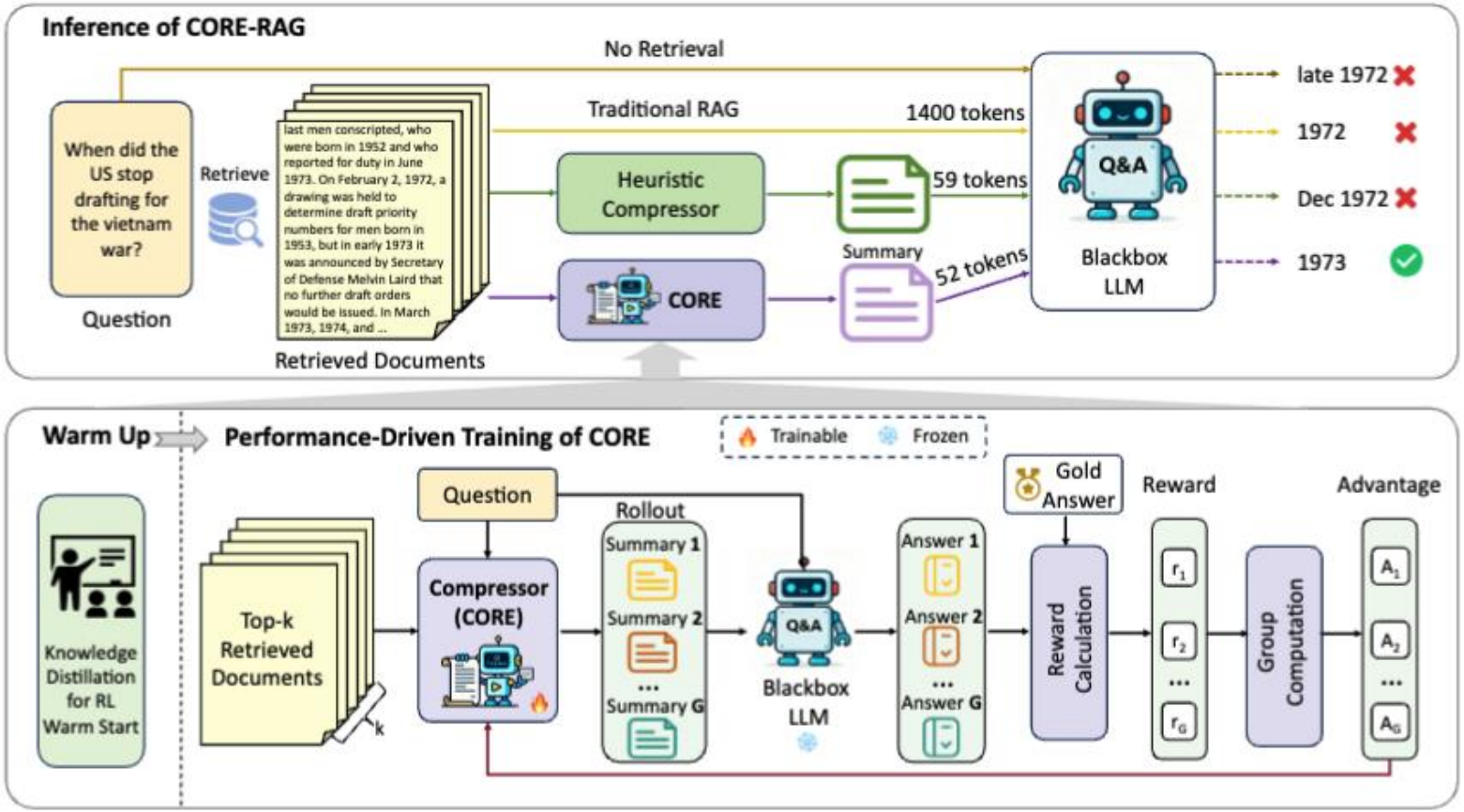


Figure 2. Overview of our method CORE. The upper section illustrates the inference pipeline. The lower section depicts the training framework for the compression model.

Main Result

Table 1. QA results on four benchmarks with Qwen2.5-14B-Instruct as the LLM (M). Both our compressor (CORE) and all trainable compression baselines (RECOMP, NoiseFilter-IB, LongLLMLingua, QGC) are trained using Qwen2.5-1.5B-Instruct to ensure a fair comparison. The reported token counts represent the length of in-context documents, excluding few-shot examples.

	NQ			TriviaQA			HotpotQA			2WikiMultihopQA		
	EM	**F1**	**# tok**	**EM**	**F1**	**# tok**	**EM**	**F1**	**# tok**	**EM**	**F1**	**# tok**
No Retrieval	21.36	30.97	0	53.23	59.98	0	21.05	29.48	0	26.11	29.51	0
RAG without compression (full-context)												
Top 1 Document	34.46	44.41	142	60.82	68.70	143	29.20	38.93	147	26.79	31.87	153
Top 3 Documents	37.78	48.45	427	62.60	71.02	430	31.64	41.87	442	27.89	33.58	460
Top 5 Documents	38.03	49.16	712	64.10	72.48	715	32.99	43.69	737	29.64	35.21	766
Top 10 Documents	38.67	50.03	1425	64.40	72.92	1428	33.95	44.88	1471	31.04	36.75	1531
Compression of top 5 documents												
BM25	25.23	36.47	37	55.36	63.90	39	24.18	35.73	71	25.42	30.29	68
Qwen2.5-1.5B-Instruct	31.94	43.03	36	57.99	66.70	30	27.36	37.47	33	25.93	31.18	32
DeepSeek-V3 (671B)	37.73	50.39	54	64.13	73.20	50	33.59	44.83	48	27.99	32.67	92
RECOMP-Abs (1.5B)	34.18	46.26	58	60.31	68.50	53	28.96	39.95	56	30.25	36.73	52
RECOMP-Ext (1.5B)	33.84	46.05	56	60.18	68.39	48	29.93	41.09	45	30.78	37.07	51
NoiseFilter-IB (1.5B)	35.15	45.94	48	59.51	68.15	35	27.97	38.62	38	27.85	34.69	40
LongLLMLingua (1.5B)	33.65	43.15	152	58.96	66.82	148	28.03	38.49	149	29.37	33.62	153
QGC (1.5B)	36.23	45.88	49	61.02	68.45	47	29.16	40.05	45	31.14	36.83	51
CORE (1.5B)	**41.02**	**50.40**	**46**	**65.63**	**72.55**	**32**	**33.67**	**45.06**	**36**	**36.72**	**42.05**	**49**
Generalization to top 10 documents (with the compressor trained on top 5 documents)												
BM25	25.91	36.88	38	55.28	63.16	37	23.49	35.01	68	25.61	30.54	65
Qwen2.5-1.5B-Instruct	32.94	44.84	40	58.45	67.31	33	28.17	38.48	36	26.22	31.57	34
DeepSeek-V3 (671B)	37.79	51.07	56	65.29	74.45	53	34.62	45.69	50	29.00	34.64	40
RECOMP-Abs (1.5B)	34.40	46.93	59	61.42	69.88	52	31.54	42.92	52	31.98	38.16	49
RECOMP-Ext (1.5B)	33.96	46.34	60	61.03	69.51	50	31.92	43.18	55	32.52	38.87	44
NoiseFilter-IB (1.5B)	35.36	46.24	50	59.92	68.32	38	28.21	38.83	38	28.63	35.16	42
LongLLMLingua (1.5B)	33.78	43.37	154	59.17	66.97	150	28.33	38.95	148	29.62	34.11	151
QGC (1.5B)	36.03	45.62	50	61.23	68.74	49	29.12	39.63	46	31.71	37.52	50
CORE (1.5B)	**41.88**	**51.26**	**52**	**66.76**	**73.64**	**33**	**34.68**	**46.35**	**37**	**37.99**	**43.28**	**48**

Table 3. QA results on four benchmarks with Llama-3.1-8B-Instruct as the LLM (M). We report the zero-shot transfer performance on this unseen LLM.

	NQ			TriviaQA			HotpotQA			2WikiMultihopQA		
	EM	**F1**	**# tok**	**EM**	**F1**	**# tok**	**EM**	**F1**	**# tok**	**EM**	**F1**	**# tok**
No Retrieval	24.04	34.91	0	55.64	62.57	0	19.93	27.75	0	27.64	31.18	0
RAG without compression												
Top 1 Document	33.80	44.06	142	59.17	67.50	143	27.95	37.49	147	28.41	33.43	153
Top 3 Documents	36.87	47.81	427	61.13	70.06	430	30.17	40.71	442	28.67	34.23	460
Top 5 Documents	37.65	48.87	712	62.26	71.04	715	31.44	42.16	737	29.43	35.18	766
Top 10 Documents	38.12	49.93	1425	63.95	72.71	1428	32.19	42.62	1471	30.45	36.04	1531
Compression of top 5 documents												
Qwen2.5-1.5B	32.60	44.21	36	56.76	65.77	30	26.86	36.90	33	25.45	30.88	32
DeepSeek-V3 (671B)	37.56	50.11	54	62.52	72.34	50	33.05	44.25	48	28.64	33.87	92
RECOMP-Abs (1.5B)	33.41	45.50	58	58.50	67.37	53	28.85	39.76	56	31.63	37.81	52
RECOMP-Ext (1.5B)	33.12	45.06	60	57.98	66.84	55	29.03	40.04	52	31.85	38.02	55
CORE (1.5B)	**40.72**	**50.00**	**46**	**64.08**	**71.13**	**32**	**32.17**	**43.71**	**36**	**35.99**	**41.42**	**49**
Generalization to top 10 documents (with the compressor trained on top 5 documents)												
Qwen2.5-1.5B	32.88	44.66	40	57.44	66.56	33	27.31	37.31	36	25.80	31.30	34
DeepSeek-V3 (671B)	37.49	51.28	56	63.79	73.80	53	34.24	45.35	50	31.45	37.09	40
RECOMP-Abs (1.5B)	34.18	46.80	59	59.69	68.89	52	30.17	41.42	55	33.61	39.78	44
RECOMP-Ext (1.5B)	34.06	46.55	60	59.33	68.71	50	30.52	41.98	55	33.52	39.42	44
CORE (1.5B)	**41.77**	**51.27**	**52**	**65.25**	**72.45**	**33**	**33.25**	**45.09**	**37**	**37.59**	**42.87**	**48**

Table 5. Comparison results on the LongBench benchmark, specifically focusing on the three Multi-Doc QA tasks.

	LongBench					
	MuSiQue		HotpotQA		2WikiMultihopQA	
	F1 score	#token	F1 score	#token	F1 score	#token
No Context	26.49	0	51.13	0	43.95	0
Full Context	38.41	11214	62.22	9151	57.97	4887
LongLLMLingua	33.18	983	57.69	907	53.26	489
CORE	**38.94**	**129**	**63.58**	**126**	**59.73**	**108**

Lossless Compression

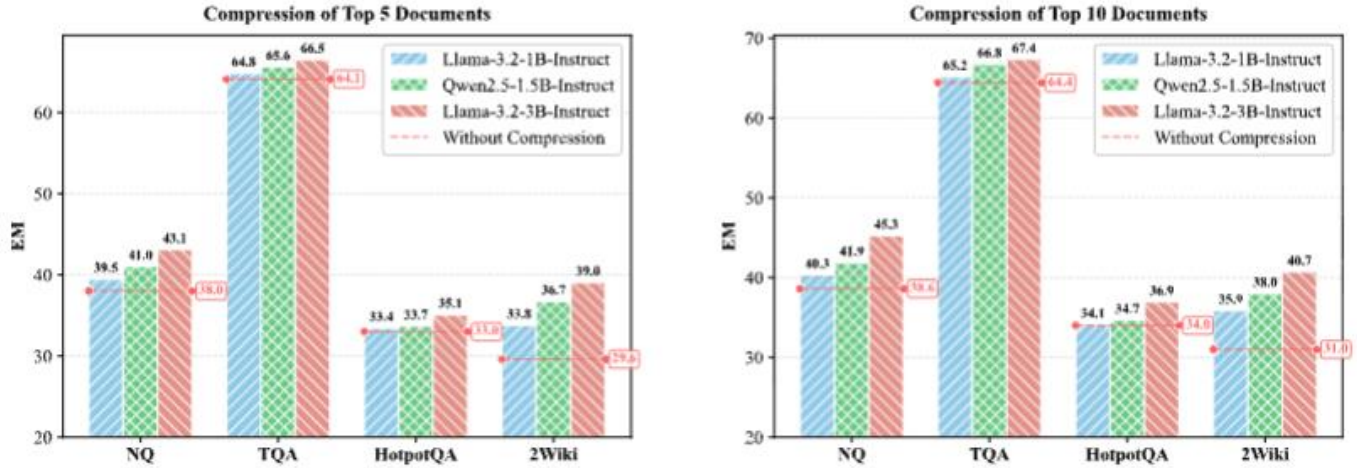
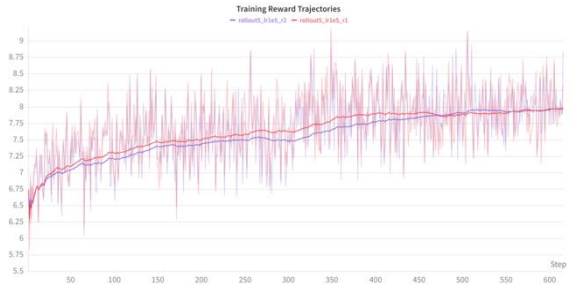


Figure 5. Robustness and Scalability: Consistent effectiveness of the proposed method across varying compressor backbones (LLaMA vs. Qwen) and model scales (1B to 3B). More detailed results are provided in Table 7 and Table 8 in the Appendix.

Table 2. Ablation study on all datasets.

Dataset	Metric	w/o distillation	w/o RL	CORE
NQ	EM	36.37	34.18	**41.02**
	F1	46.91	46.26	**50.40**
TQA	EM	65.23	60.31	**65.63**
	F1	72.41	68.50	**72.55**
HotpotQA	EM	32.01	28.96	**33.67**
	F1	42.73	39.95	**45.06**
2Wiki	EM	31.40	30.25	**36.72**
	F1	36.89	36.73	**42.05**

- **Lossless Compression**
- **Scaling Law**
  : Llama-3.2-1B ＜ Qwen2.5-1.5B ＜ Llama-3.2-3B
- **Robustness**



- **w/o distillation:**
  Distillation 건너뛰고 바로 RL
- **w/o RL:**
  강화 학습 단계를 제외하고 지식 증류만 수행
- **CORE:**
  본 논문이 제안한 전체 프레임워크(지식 증류 + 강화 학습)를 모두 적용

Thank you