Dynamic Multi-Grained Retrieval-Augmented Generation for Multi-Hop QA

Yu Pei[1,2,3,4], Mieradilijiang Maimaiti[1,2,3,4,*], Yi Chen[1,2,3,4], Wu Le[5] , Zhuofei Xie[5] , Jiawei Chen[5]

[1] School of Computer Science and Technology, Xinjiang University
[2] Xinjiang Laboratory of Multi-Language Information Technology, Xinjiang University
[3] Xinjiang Multilingual Information Technology Research Center
[4] Joint International Research Laboratory of Silk Road Multilingual Cognitive Computing No. 777, Huarui Street, Shuimogou District, Urumqi 830017, Xinjiang, China
[5] Integrated Laboratory for Space, Air, and Ground Systems

Overview & Motivation

Research Gap

Existing RAG pipelines usually use fixed chunking, fixed retrieval, and simple context packing.

However, multi-hop QA requires different evidence granularities for different questions.

Why It Is Challenging

Sentence chunks are precise but may fragment evidence chains, while document chunks provide coverage but introduce noise.

Under a limited token budget, naive top-k retrieval often fails to keep complete supporting evidence.

Our Goal

We propose DMG-RAG, a query-adaptive multi-grained RAG framework. It dynamically allocates retrieval quotas across sentence/paragraph/document indices and selects compact evidence under a fixed budget.

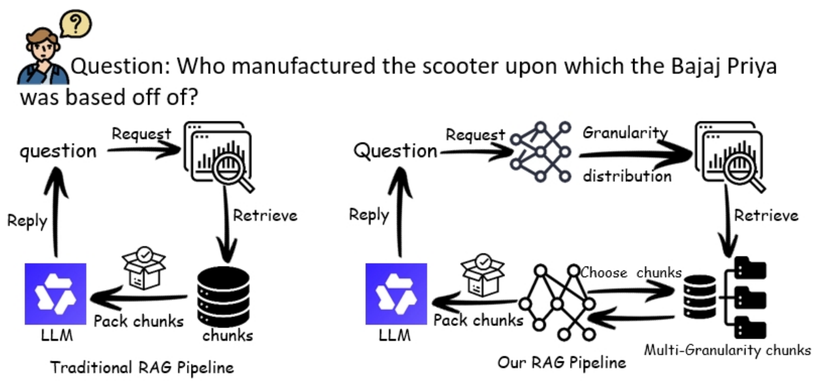


Fig. 1: Pipeline comparison between a traditional retrieve then-read RAG pipeline and our Dynamic Multi-Grained RAG pipeline.

METHOD

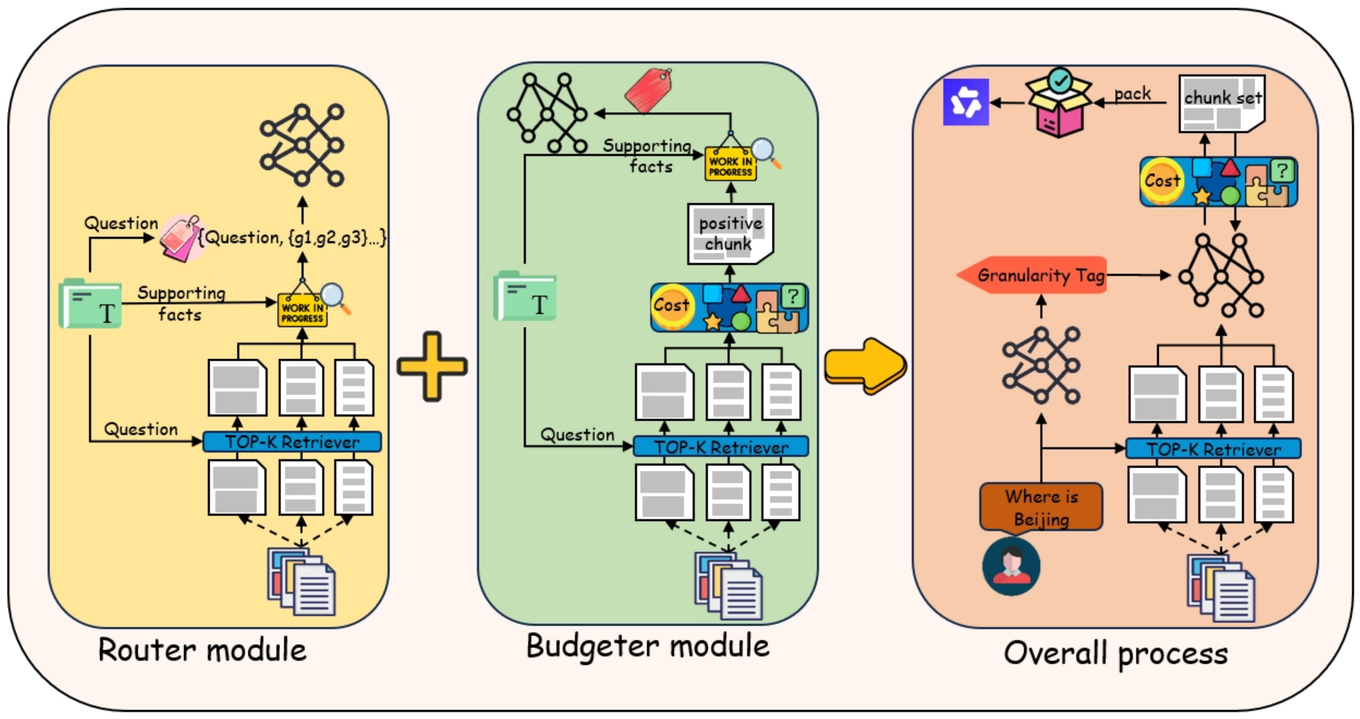


Fig. 2. Overview of DMG-RAG. Router allocates retrieval quotas, and Budgeter performs budget-aware packing.

Multi-Grained Retrieval

- Build three aligned indices: sentence, paragraph, and document.
- Retrieve candidates from multiple granularities for each query.
- Optionally rerank top candidates with a cross-encoder.

Router

- Predicts a granularity distribution for each question.
- Converts it into top-k retrieval quotas.
- Shapes a query-adaptive candidate pool.

Budgeter

- Scores candidate chunks using relevance, length, and redundancy signals.
- Selects high-utility chunks under token budget B.
- Uses deterministic greedy packing for stable generation.

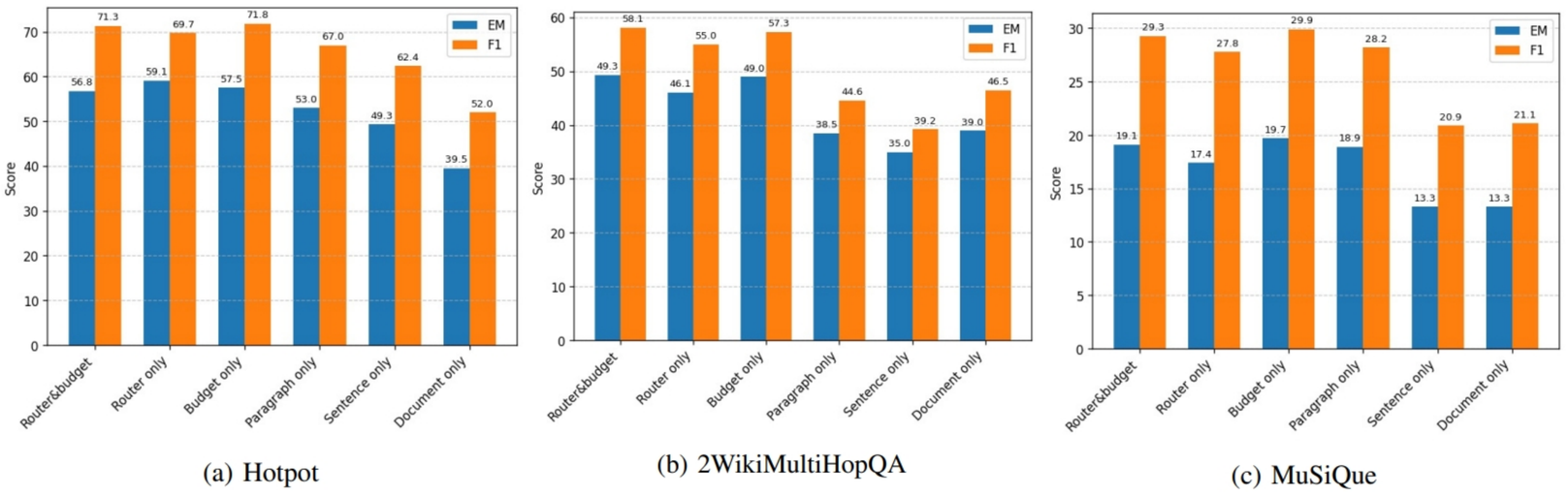
RESULTS

Methods	HotpotQA		2WikiMultiHopQA		Average	
	EM	F1	EM	F1	EM	F1
Direct(qwen2.5-14B) [7]	20.6	27.4	23.2	25.49	21.9	26.5
Naive RAG [1]	53.1	67.1	39	46.5	46.05	56.8
MoGRAG [6]	40.4	52.37	19	23.61	29.7	37.99
GenGroundRAG [27]	36.59	45.91	21.2	26.42	28.9	36.17
Ours	**56.8**	**71.3**	**49.3**	**58.1**	**53.05**	**64.7**

Table. 1. Main results on HotpotQA and 2WikiMultiHopQA.

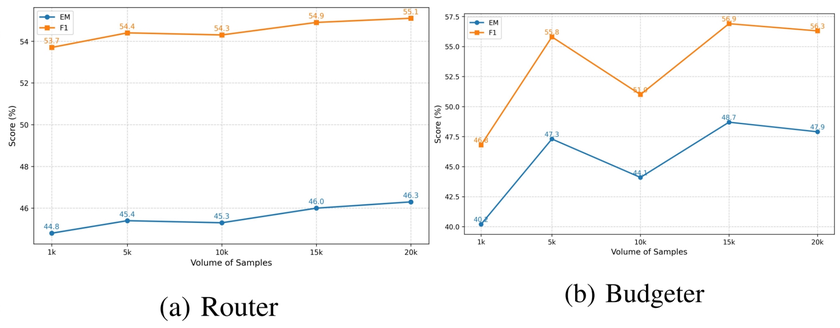
Methods	MuSiQue	
	EM	F1
Direct [7]	5	13.6
Naive RAG [1]	18.7	28.3
MoGRAG [6]	6.6	12.29
GenGroundRAG [27]	10.87	15.1
Ours	**19.1**	**29.3**

Table. 2. Main results on Musique.



(a) Hotpot (b) 2WikiMultiHopQA (c) MuSiQue

Fig. 3. Ablation results across three multi-hop QA benchmarks.



(a) Router (b) Budgeter

Fig. 4. Sensitivity to training size and training steps.

CONCLUSION

- We propose DMG-RAG for query-adaptive multi-grained RAG.
- Router dynamically allocates retrieval quotas across sentence/paragraph/document indices.
- Budgeter selects compact and complementary evidence under a fixed context budget.
- Experiments on three multi-hop QA benchmarks show consistent improvements.

Acknowledgement: Supported by the Xinjiang “Tianchi Talent” Recruitment and Introduction Program.

* **Corresponding author:** miradeljan51@xju.edu.cn