DMG-RAG

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

Yu Pei et al. | Xinjiang University

THE CORE IDEA

Route

retrieval quota across
3 granularities

Budget

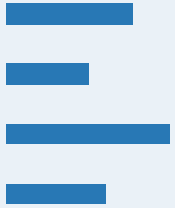
select complementary
evidence
under a fixed token budget

+11.6 F1 over Naive RAG on 2WikiMultiHopQA

MOTIVATION

One fixed chunk size cannot serve every multi-hop question.

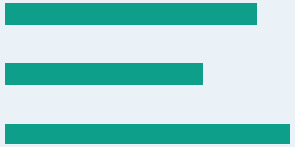
SENTENCE



Precise

but evidence chains fragment

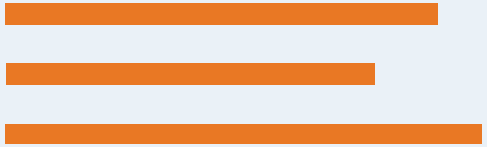
PARAGRAPH



Balanced

but the best span varies by query

DOCUMENT



Complete

but noise consumes the budget

Query heterogeneity turns chunk granularity into a per-question decision.

Source: DMG-RAG paper

KEY IDEA

DMG-RAG separates candidate formation from evidence composition.

1 ROUTER

Query-level decision

Allocate stage-1 top-k quota

sentence · paragraph · document

Output: an adaptive candidate pool

→

2 BUDGETER

Chunk-level decision

Score utility and pack evidence

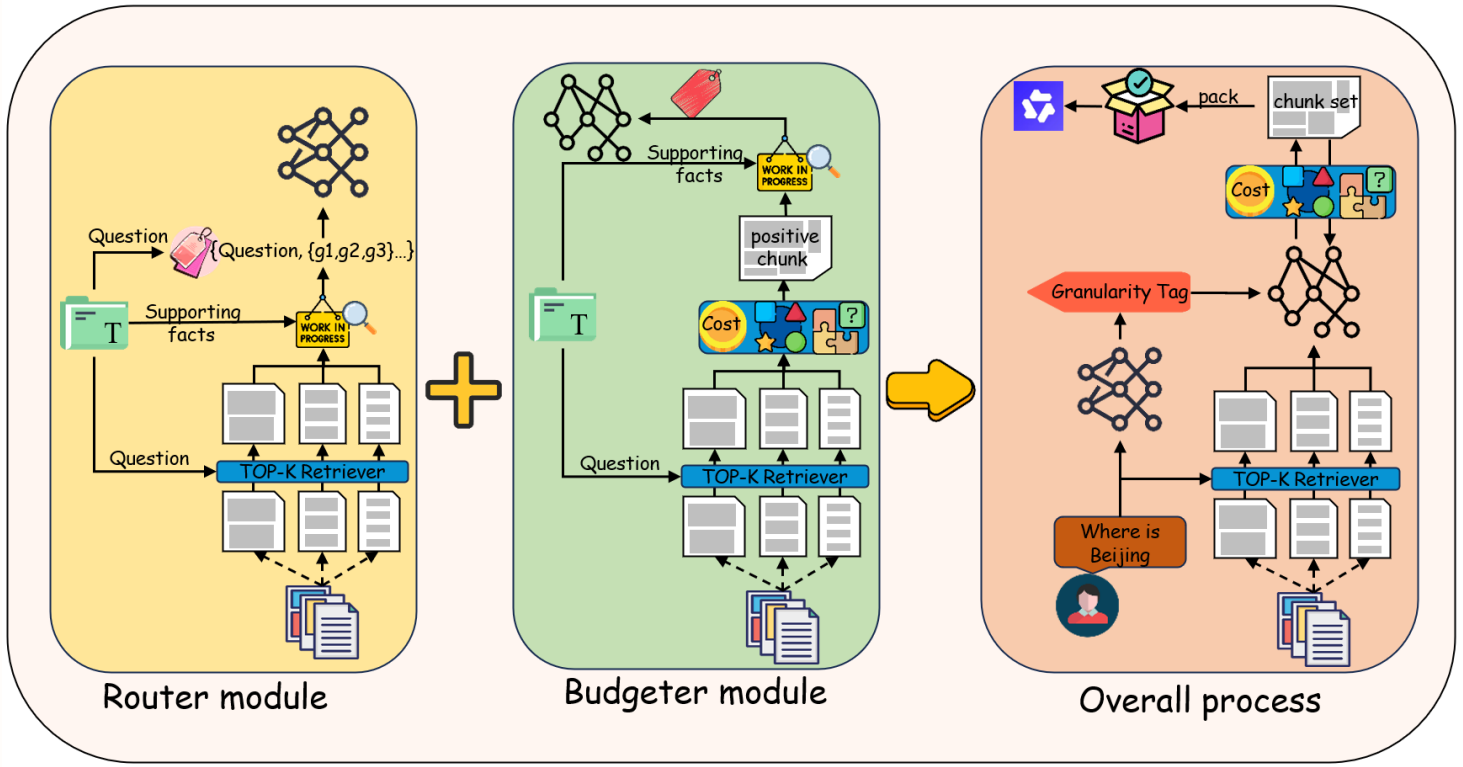
relevance · length · redundancy

Output: a compact complementary context

Source: DMG-RAG paper

ARCHITECTURE

The pipeline adds two lightweight modules around standard retrieval.



1 Route quotas

2 Retrieve candidates

3 Score & pack

4 Generate answer

Source: DMG-RAG paper

ROUTER

The Router reshapes retrieval capacity, not individual rankings.

QUERY

q

query embedding

+ question length
+ normalized budget

lightweight MLP + softmax

base quota · prior mixing · sentence cap

→

Sentence — **k_sent**

Paragraph — **k_para**

Document — **k_doc**

Constraint: k_sent + k_para + k_doc = K

Source: DMG-RAG paper

The Budgeter spends tokens on utility, not rank alone.

CANDIDATE POOL

S	**utility 0.92**	42 tokens
P	**utility 0.87**	116 tokens
D	**utility 0.81**	420 tokens
S	**utility 0.77**	38 tokens
P	**utility 0.73**	104 tokens

→

UTILITY SCORER

s(c, q)

query embedding
retrieval / rerank score
granularity id
token length

→

GREEDY PACKING

B = fixed token budget

selected: complementary chunks

skip if over budget
penalize redundancy
deterministic output

Learn scores. Keep selection deterministic.

Source: DMG-RAG paper

EVALUATION

A controlled setup isolates the value of retrieval and evidence selection.

BENCHMARKS

HotpotQA
2WikiMultiHopQA
MuSiQue

1,500 questions each · fixed seed

MODELS

Qwen2.5-14B-Instruct
bge-large-en-v1.5
bge-reranker-large

Same generator across methods

METRICS

Exact Match (EM)
Token-level F1
Deterministic decoding

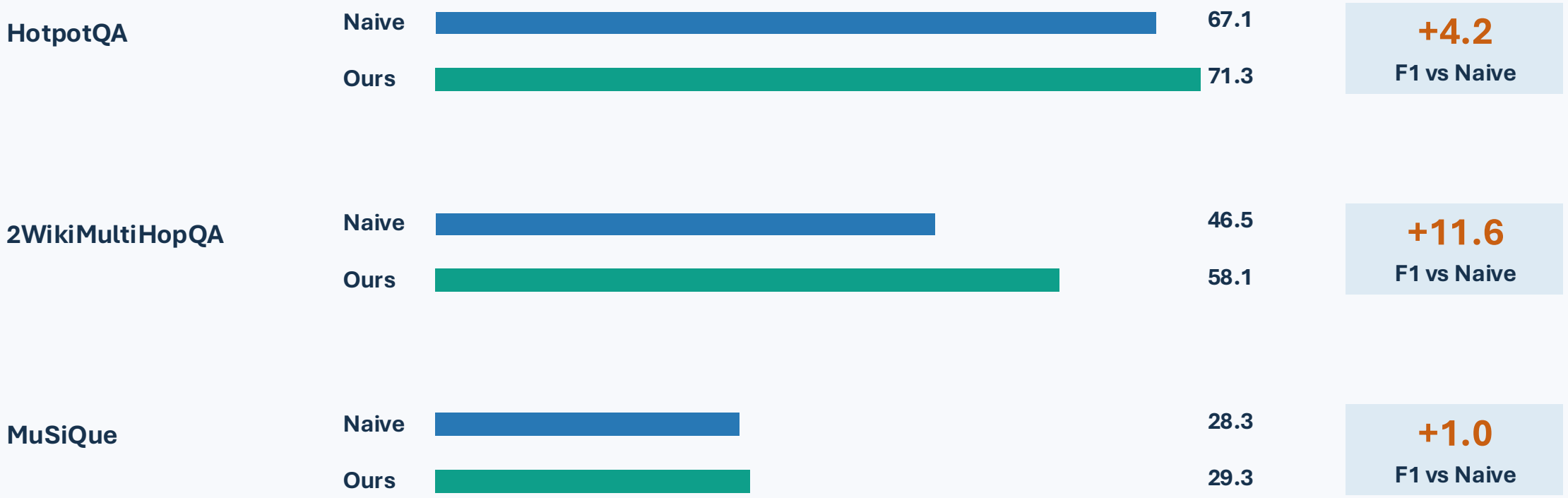
Higher is better

Baselines: Direct · Naive RAG · MoG-RAG · GenGroundRAG

Source: DMG-RAG paper

MAIN RESULTS

DMG-RAG gains most where cross-document evidence is hardest to assemble.

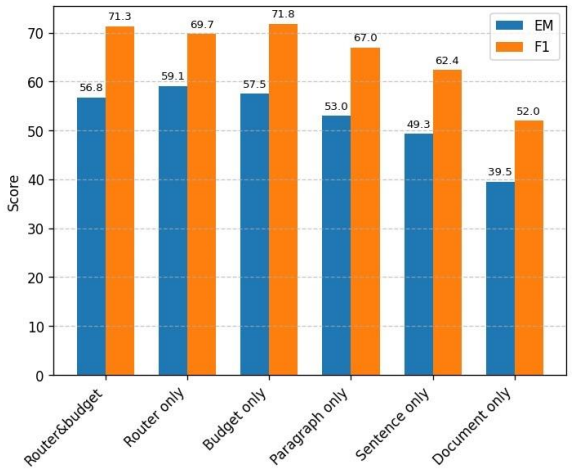


Average on HotpotQA + 2Wiki: 64.7 F1 | 53.05 EM

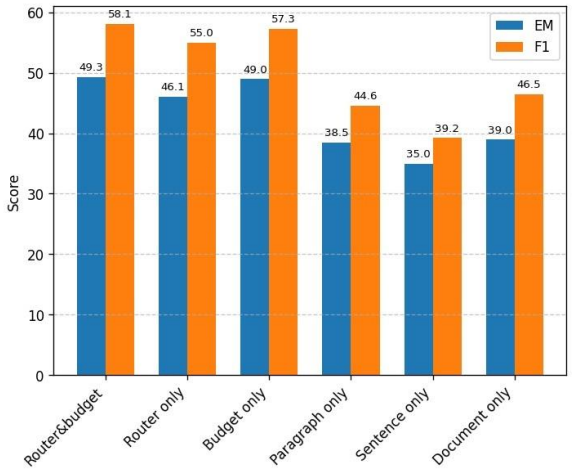
Source: DMG-RAG paper

ABLATION & TAKEAWAY

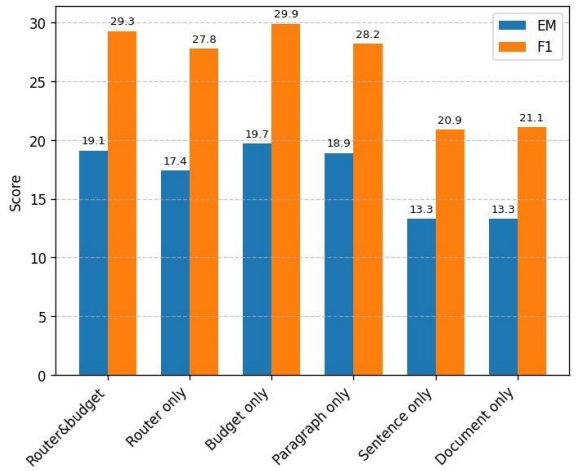
Set-level selection drives performance; routing makes the gains consistent.



HotpotQA



2WikiMultiHopQA



MuSiQue

Budgeter-only is usually stronger than Router-only.

Router + Budgeter is the most consistent combination.

Single granularity leaves either gaps or noise.

DMG-RAG coordinates evidence units and evidence composition under one fixed budget.

Source: Figure 3 and Section IV-D

Why are both the Router and the Budgeter necessary?

Q1 Why not select only one granularity?

Different queries need different evidence units. Sentence chunks are precise but can break evidence chains. Document chunks preserve context but introduce noise. The Router creates a query-specific mixture instead of forcing one fixed choice.

Q2 Why separate routing from packing?

They solve different problems. Routing controls candidate recall across granularities. Packing controls evidence quality inside the token budget. Separating them keeps both decisions lightweight, interpretable, and easy to ablate.

How reliable and fair are the experimental comparisons?

Q3 Is the comparison with baselines fair?

All generation-based methods use the same Qwen2.5-14B-Instruct generator. Retrieval settings, deterministic decoding, evaluation subsets, and random seeds are controlled so that the main difference is evidence construction.

Q4 Why is the gain smaller on MuSiQue?

MuSiQue places stronger pressure on initial recall. If a key fact is absent from the candidate pool, the Budgeter cannot recover it. This result identifies retrieval recall, rather than packing, as the next bottleneck.

What are the main limitations and next steps?

Q5 What is the main limitation?

The Router and Budgeter use weak supervision generated on the 2Wiki training split. This is efficient, but label quality depends on the oracle search and automatic answerability signals. Broader training sources may improve generalization.

Q6 What will you improve next?

We will strengthen candidate recall through retriever adaptation or query expansion, refine supervision for routing and packing, and test dynamic corpora, more datasets, and different generators.