Design of an Automated Framework for Evaluating the Performance of Multimodal RAG Models

Kyungwon Kim
Information & Media Research Center
Korea Electronics Technology *Institute*
Seongnam, Republic of Korea
kwkim@keti.re.kr

Jongjin Jung
Information & Media Research Center
Korea Electronics Technology Institute
Seongnam, Republic of Korea
mozzalt@keti.re.kr

Jong-Bin Park
Information & Media Research Center
Korea Electronics Technology Institute
Seongnam, Republic of Korea
jpark@keti.re.kr

Ju-Young Kim
Information & Media Research Center
Korea Electronics Technology Institute
Seongnam, Republic of Korea
jykim@keti.re.kr

***Abstract*—Retrieval-Augmented Generation (RAG) that combines heterogeneous modalities—text, image, audio, and video—is increasingly used to deliver reliable answers in real-world applications. However, collecting, curating, and building evaluation sets by domain is costly and time-consuming, and manual evaluation pipelines do not reflect production conditions well and do not scale. This paper proposes an automated framework that measures the performance of multimodal RAG models objectively, reproducibly, and fairly by integrating modules for data preprocessing, auto-benchmark generation, API-based response collection, claim-level evaluation, and result reporting. The design features standardized preprocessing and indexing for multimodal corpora, requirement-controlled auto-benchmarking using LLMs, an API-only evaluation protocol that prevents direct data access, a claim-based metric suite extended to multimodality, and reporting with versioning, execution logs, and dashboards. The framework supports fair comparisons across domains and models, and provides actionable diagnosis for model iteration and quality control.**

Keywords—Multimodal RAG, Auto-Benchmarking, Claim-based Metrics, API-based Evaluation, Reporting

I. INTRODUCTION

Large Language Models (LLMs) show strong performance across knowledge-intensive tasks, including open-domain question answering, but suffer from recency limitations and hallucination, which challenge their standalone reliability. Retrieval-Augmented Generation (RAG) mitigates these issues by combining external knowledge retrieved at inference time with generation, and has spread rapidly[1]. In practice, there is growing demand for multimodal RAG that leverages evidence beyond text (e.g., images, speech, and video). Yet standardized and automated evaluation methodology for multimodal RAG remains underdeveloped. Building domain-specific evaluation sets by hand is expensive, slow, susceptible to bias, and often misaligned with real usage scenarios.

We introduce a framework that automates the full evaluation pipeline for multimodal RAG. The system unifies requirement-driven auto-benchmark generation, API-only response collection, and fine-grained claim-level diagnostics. The end-to-end pipeline follows a consistent flow(corpus upload → requirement analysis & specification → benchmark generation → API-based RAG connectivity → claim evaluation → reporting) and refines data structures and metrics to strengthen generality and extensibility for multimodal settings.

II. RELATED WORK

RAG's structure and benefits have been widely validated. Recent studies on RAG benchmarking and automatic data generation demonstrate both the efficiency and controllability of LLM-based benchmark construction[2-9]. Auto-generated Q&A benchmarks tailored for enterprise settings show advantages for reproducibility and maintainability in production. RAGChecker[10] introduced claim extraction from answers and references to diagnose accuracy, hallucination, and context use at the claim level.

Building on this line of work, our framework proposes a claim-based metric suite (Precision, Recall, Context Precision, Claim Recall, Context Utilization, Noise Sensitivity, Hallucination, Self-Knowledge, Faithfulness) and extends it to multimodality. To ensure fairness, we adopt an API-only protocol that blocks direct access to the evaluation data and standardizes request/response schemas. We generalize both theory and practice to the multimodal domain and provide a reference design that spans architecture, data modeling, and operational strategy.

III. DESIGN GOALS AND PRINCIPLES

Functional requirements include automated preprocessing-to-reporting, support for text and image (extensible to speech and video) embedding and indexing, requirement-controlled benchmark generation, API-only evaluation without data exposure, and claim-level metrics. Non-functional requirements include reproducibility (parameter and result versioning), security (PII masking and access control), observability (metrics and logs), and maintainability (modular boundaries and standard interfaces).

Design principles: (1) separation of concerns across collection, indexing, generation, scoring, and reporting; (2) declarative configuration via structured data (e.g., YAML/JSON) to enable portability and automation; (3) deterministic defaults(fixed seeds, search parameters, prompt versions) to ensure reproducibility; (4) human-in-the-loop

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2710017875, Development of multimodal data input-based search augmentation generation technology)

979-8-3315-7896-1/26/$31.00 ©2026 IEEE

inspection and error labeling without breaking the automated pipeline.

IV. ARCHITECTURE OF THE AUTOMATED EVALUATION FRAMEWORK

As illustrated by the framework diagram in the original manuscript (Fig. 1), the system comprises four layers: Data, Services, Orchestration, and Reporting. The data layer manages raw corpora, modality metadata, embedding indexes, and access policies for independently deployable RAG targets. The services layer includes preprocessing/embedding, benchmark generation, RAG connectors, and claim-based evaluation—exposed via standardized gRPC/REST interfaces. The orchestration layer uses a state-machine workflow manager and scheduler for distribution, retries, and monitoring. The reporting layer provides dashboards for aggregate metrics, component-wise breakdown (retriever vs. generator), and comparisons by modality and difficulty.

Fig. 2 illustrates each layer's constituent submodules and their detailed inter-module interactions.

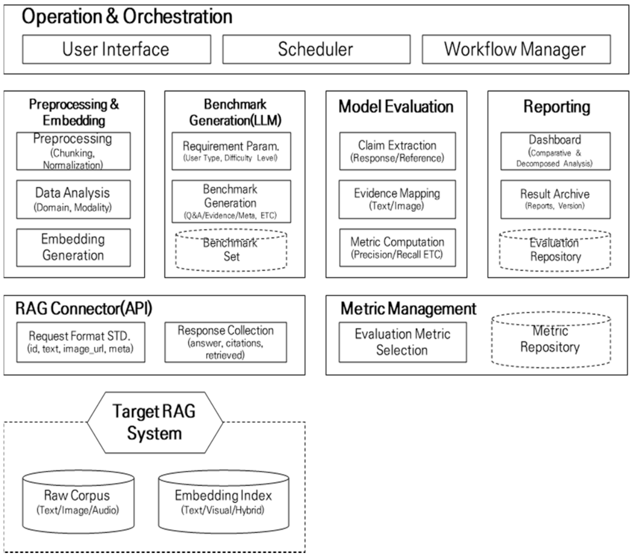
Operation & Orchestration
User Interface
Scheduler
Workflow Manager
Preprocessing & Embedding
Preprocessing (Chunking, Normalization)
Data Analysis (Domain, Modality)
Embedding Generation
Benchmark Generation(LLM)
Requirement Param. (User Type, Difficulty Level)
Benchmark Generation (Q&A/Evidence/Meta, ETC)
Benchmark Set
Model Evaluation
Claim Extraction (Response/Reference)
Evidence Mapping (Text/Image)
Metric Computation (Precision/Recall ETC)
Reporting
Dashboard (Comparative & Decomposed Analysis)
Result Archive (Reports, Version)
Evaluation Repository
RAG Connector(API)
Request Format STD. (id, text, image_url, meta)
Response Collection (answer, citations, retrieved)
Metric Management
Evaluation Metric Selection
Metric Repository
Target RAG System
Raw Corpus (Text/Image/Audio)
Embedding Index (Text/Visual/Hybrid)

Fig.1. *Automated evaluation framework architecture*

A. *Multimodal Data Preprocessing and Indexing*

Text is chunked with respect to topic and length, bound with metadata, and embedded to support hybrid retrieval. Images are processed to extract captions and semantic tags (objects, relations, scenes), and visual embeddings are joined with text embeddings for combined indexing[11-14]. Audio/video are normalized into searchable form via STT, shot segmentation, and key-frame extraction. All artifacts are versioned for traceability.

B. *Auto-Benchmark Generation*

Benchmarks are generated from corpus analysis and explicit user requirements encoded in prompts. Parameters include user type, question difficulty, modality mix, answer format, and evaluation focus (fact recall, reasoning, explanation). The LLM produces question–answer–evidence triplets aligned with these requirements; quality control applies automatic filters for de-duplication, difficulty balancing, bias checks, and hard negative insertion.

C. *API-Based Evaluation*

To ensure fairness, the target RAG system is evaluated exclusively through a standardized RAG Connector API, and direct access to the benchmark corpus is not permitted. The request message comprises a question identifier, the query text, modality resources (e.g., image URIs), retrieval parameters (e.g., top-k), and execution metadata (e.g., run_id, benchmark_id). The corresponding response message contains the model's output, citations (document identifiers and spans), retrieval results (document identifiers, scores, and modalities), and latency. The request–response specification is standardized to guarantee interoperability across heterogeneous RAG implementations.

D. *Claim Extraction and Multimodal Metrics*

The answer and reference are decomposed into claims at the sentence or phrase level. Each claim is labeled as true, false, or insufficient based on its consistency with the gold answer, the retrieved context, and visual evidence (e.g., captions, object tags, or regions). The claim-based evaluation metrics (Precision, Recall, Context Precision, Claim Recall, Context Utilization, Noise Sensitivity, Hallucination, Self-Knowledge, and Faithfulness) are summarized in Table 1. For multimodal evaluation, we additionally introduce auxiliary metrics such as Visual Utilization (degree of reliance on visual evidence) and Cross-modal Consistency (absence of contradictions between text and images), as presented in Table 2. Aggregation is performed at the levels of question, category, modality, difficulty, and run, and results are reported with confidence intervals and weighted means[15].

TABLE I. Claim-based evaluation metrics

Metric	Definition / Meaning
Precision	The proportion of answer claims that are factually correct. A claim is counted as correct only if its entities, relations, numbers/units, and temporal qualifiers are accurate
Recall	The proportion of gold (reference) claims that the system reproduces correctly in its answer.

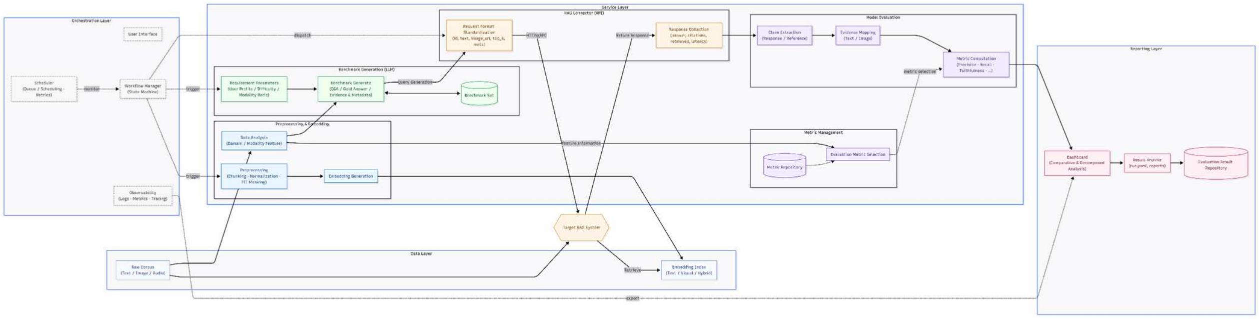
Orchestration Layer
User Interface
Scheduler (Queue / Scheduling · Retries)
monitor
Workflow Manager (State Machine)
trigger
trigger
Observability (Logs · Metrics · Tracing)
Service Layer
RAG Connector (API)
dispatch
Request Format Standardization (id, text, image_uri, top_k, meta)
HTTP/gRPC
Return Response
Response Collection (answer, citations, retrieved, latency)
Benchmark Generation (LLM)
Requirement Parameters (User Profile / Difficulty / Modality Ratio)
Benchmark Generate (Q&A / Gold Answer / Evidence & Metadata)
Query Generation
Benchmark Set
Preprocessing & Embedding
Data Analysis (Domain / Modality Feature)
Preprocessing (Chunking · Normalization · PII Masking)
Embedding Generation
feature information
Model Evaluation
Claim Extraction (Response / Reference)
Evidence Mapping (Text / Image)
Metric Computation (Precision · Recall · Faithfulness · ...)
metric selection
Metric Management
Metric Repository
Evaluation Metric Selection
Reporting Layer
Dashboard (Comparative & Decomposed Analysis)
Result Archive (run.yaml, reports)
Evaluation Result Repository
Target RAG System
Data Layer
Raw Corpus (Text / Image / Audio)
Retrieve
Embedding Index (Text / Visual / Hybrid)
export

Fig.2. *Automated evaluation framework Top-level pipeline (Layers, Modules & Interaction)*

	Paraphrases that preserve meaning count as correct matches
Context Precision	A retrieval-quality measure: among all retrieved chunks, the share that are truly relevant to the gold answer
Claim Recall	An upper bound on answerability from retrieval: the share of gold claims that can be found somewhere in the retrieved set (regardless of whether the generator uses them)
Context Utilization	How much the generator actually relies on retrieved evidence when composing its answer. A claim is considered utilized if it is explicitly backed by at least one retrieved span (via citation or alignment)
Noise Sensitivity	Robustness to irrelevant (noisy) retrieval. Report the change in a target metric (e.g., Precision/Recall/F1) when distractor chunks are injected: more negative change indicates higher vulnerability
Hallucination	The share of answer claims that are incorrect and unsupported by the retrieved context(i.e., fabricated or contradicted by evidence)
Self-Knowledge	The extent to which the model can produce correct claims without retrieval
Faithfulness	The degree to which answer claims are entailed by (and not contradicted by) the retrieved evidence. Correct world knowledge that conflicts with cited context is treated as unfaithful

TABLE II. Additional Evaluation Metrics for Multimodal RAG

Metric	Definition / Meaning
Visual Utilization	Degree to which the answer materially uses the provided visual evidence (image/video) rather than relying only on prior text or world knowledge
Cross-modal Consistency	Absence of contradictions between textual content and the visual evidence. Evaluate each answer for conflicts such as wrong objects ("a red car" vs. blue car), wrong counts, swapped relations ("A left of B" vs. right), or temporal mismatches (claiming "door open" while frames show closed)
Visual-Grounded Faithfulness	Share of answer claims that are directly supported (entailed) by visual evidence. A claim is counted if at least one region/frame/segment entails it via detectors/segmenters/pose/ASR-OCR (for charts, text in image), or an LLM-VQA verifier

E. *Evaluation Results Reporting*

The reporting system supports cross-model comparison and component-wise decomposition (retriever vs. generator), and automatically produces diagnostic cards for each item—including hallucination sources, unused context, and cross-modal mismatches. In a dashboard interface, both aggregate and category-level metrics are visualized, analogous to a standard evaluation-results reporting UI. All evaluation runs are fully versioned and accompanied by detailed condition and result logs, ensuring reproducible outcomes.

V. CONCLUSION

The principal strengths of the proposed framework are fairness, reproducibility, and extensibility. By enforcing an API-centric evaluation protocol, the system eliminates direct data access, thereby reducing opportunities for cheating. Versioning of execution parameters and evaluation outputs enables like-for-like regression testing even after model updates. Modularization and standardized interfaces facilitate the seamless incorporation of new domains and modalities. Moreover, claim-level diagnostics disentangle failure modes of the retriever and generator, and allow early detection of multimodal-specific errors such as text–image inconsistencies.

There are, however, limitations. Because claim extraction partially relies on LLMs, domain-specific expressions can be misjudged. Improving evidence alignment for video and audio further requires higher-order semantic matching—e.g., reasoning over object relations and scene transitions. For real-time streaming evaluations, latency and cost control remain practical challenges. Future work includes evaluating tool-using conversational agents (e.g., retrieval and code execution) and enhancing evidence–claim alignment via semantic grounding graphs.

In summary, we present an automated framework for evaluating multimodal RAG models that integrates multimodal preprocessing and indexing, LLM-based automatic benchmark generation, API-based fair evaluation, fine-grained claim-level metrics, and versioned reporting. We also systematize operational strategies that reflect the characteristics of multimodal data across diverse domains, thereby ensuring generality and scalability. The proposed framework enables fair comparison, root-cause diagnosis, and regression testing across a wide range of applications, and can accelerate the industrial adoption of multimodal RAG.

REFERENCES

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," NeurIPS, 2020, pp. 9459–9474.

[2] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking Large Language Models in Retrieval-Augmented Generation," AAAI, 2024, 38(16), pp. 17754–17762.

[3] C. Dong, Y. Shen, S. Lin, Z. Lin, and Y. Deng, "A Unified Framework for Contextual and Factoid Question Generation," IEEE TKDE, 2023, 36(1), pp. 21–34.

[4] X. L. Do, B. Zou, L. Pan, N. F. Chen, S. Joty, and A. T. Aw, "COHS-CQG: Context and History Selection for Conversational QG," arXiv:2209.06652, 2022.

[5] S. Filice, G. Horowitz, D. Carmel, Z. Karnin, L. Lewin-Eytan, and Y. Maarek, "Generating Q&A Benchmarks for RAG Evaluation in Enterprise Settings," ACL Industry Track, 2025, pp. 469–484.

[6] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-T. Yih, "Dense Passage Retrieval for Open-Domain Question Answering," EMNLP, 2020.

[7] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang, "REALM: Retrieval-Augmented Language Model Pre-Training," ICML, 2020.

[8] G. Izacard and E. Grave, "Leveraging Passage Retrieval with Generative Models for Open-Domain Question Answering (FiD)," ICLR, 2021.

[9] Izacard, G., et al., "ATLAS: Few-shot Learning with Retrieval-Augmented Language Models," arXiv preprint, 2022.

[10] D. Ru, L. Qiu, X. Hu, T. Zhang, P. Shi, S. Chang, and Z. Zhang, "RAGChecker: A Fine-Grained Framework for Diagnosing Retrieval-Augmented Generation," NeurIPS, 2024, 37, pp. 21999–22027.

[11] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, et al., "Learning Transferable Visual Models from Natural Language Supervision (CLIP)," ICML, 2021.

[12] J.-B. Alayrac, A. Miech, A. Bursuc, et al., "Flamingo: a Visual Language Model for Few-Shot Learning," NeurIPS, 2022.

[13] J. Li, D. Li, S. C. H. Hoi, and S. Savarese, "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models," arXiv preprint, 2023.

[14] Liu, H., Li, C., Wu, Q., and Lee, Y. J., "Visual Instruction Tuning (LLaVA)," arXiv preprint, 2023.

[15] Fabbri, A., Kryściński, W., McCann, B., Xiong, C., Socher, R., and Radev, D., "QAFactEval: Improved QA-based Factual Consistency Evaluation for Summarization," Proceedings of EMNLP, 2022.