Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Knowledge compression and knowledge graph-based retrieval-augmented generation for large language models in dairy cow precision feeding

Bowen Li
Hebei Agricultural University
Yongsheng Si
Hebei Agricultural University
Chendong Yang
Hebei General Station of Animal Husbandry and Improved Livestock Breeds
Hongtian Li
Chinese Academy of Agricultural Sciences
Paul Kwan
Central Queensland University
Leifeng Guo
guoleifeng@caas.cn
Chinese Academy of Agricultural Sciences
Yabin Ma
Hebei General Station of Animal Husbandry and Improved Livestock Breeds

Article

Keywords: dairy cow precision feeding, knowledge compression, large language models, retrieval-augmented generation, LightRAG, multi-judge evaluation

Posted Date: November 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11663290/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: Competing interest reported. Leifeng Guo is a member of the Editorial Board of Scientific Reports and a Guest Editor of the “Precision Livestock Farming and on-Farm AI” Collection. He is also the academic supervisor of Bowen Li. The authors declare no other competing interests.

Abstract

Knowledge for dairy cow precision feeding is dispersed across nutritional standards, scientific literature and professional resources, complicating evidence-supported question answering with large language models (LLMs). We integrated a curated domain corpus with LightRAG, an existing retrieval-augmented generation (RAG) framework combining knowledge graphs and vector retrieval. Corpus preparation comprised PDF-to-Markdown conversion, two-stage rule-based cleaning and document-level knowledge compression. Compression prioritised feeding-relevant evidence, numerical values, units and applicability conditions while retaining source identifiers, reducing cleaned literature text by 95.53%. The final corpus contained 2,925 documents and 6,992,080 non-whitespace characters, yielding 5,610 text chunks, 52,509 entities and 64,863 relations. We evaluated DeepSeek v4, Qwen 3.7 Max and GLM 5.3 with and without LightRAG on 100 frozen questions, obtaining 600 responses independently scored by ChatGPT and Kimi-k3 using the same blinded rubric. ChatGPT mean question–answer matching scores increased from 55.8 to 78.7, 43.5 to 79.2 and 65.3 to 80.2, respectively; Kimi-k3 scores increased from 39.8 to 80.6, 31.6 to 80.3 and 44.7 to 82.8. LightRAG-enhanced responses scored higher in 299 of 300 ChatGPT comparisons and 298 of 300 Kimi-k3 comparisons. These findings support curated knowledge and hybrid retrieval to improve question–answer matching for dairy cow precision feeding under the specified evaluation protocol.

Introduction

Precision feeding of dairy cows must account for physiological stage, milk yield, dry matter intake (DMI), diet composition, health status and environmental conditions. Nutritional standards and on-farm monitoring guide adjustments to feeding strategies[1]. Relevant knowledge is dispersed across nutrient requirement systems, research papers, technical guidelines and production records. These sources differ in format, terminology and the granularity of evidence. Converting this unstructured knowledge into retrievable, traceable and updatable resources is a prerequisite for domain question answering and decision support.

Large language models (LLMs) can interpret and generate text, but their parametric knowledge may omit recent or specialised information. They can also generate plausible but factually incorrect statements when evidence is insufficient[2]. For nutritional thresholds, disease risks and dietary adjustments, correctness, applicability and source traceability matter more than fluency. Supporting dairy cow precision feeding therefore requires integrating model generation with curated external knowledge, a consistent evaluation protocol and expert review.

Retrieval-augmented generation (RAG) retrieves external evidence before generating responses and can support knowledge updates and source tracing[3]. Knowledge graphs organise information within a domain as entities and relations[4,5]. GraphRAG and LightRAG integrate graph structures into retrieval, combining local entity evidence with global thematic information to inform responses[6,7]. We used

LightRAG to construct an index for dairy cow precision feeding. RAG evaluation should consider response faithfulness, relevance, retrieval quality and domain safety[8].

Language models and RAG systems have been developed for agricultural examinations, knowledge graphs and producer questions, highlighting the potential of domain knowledge enhancement[9,10,11]. However, these applications differ from dairy cow precision feeding in their tasks, information sources and risk boundaries. Dairy nutrition and feeding management therefore require source governance, test isolation and domain-specific evaluation procedures.

We developed document-level knowledge compression for dairy cow precision feeding, linking physiological stage, nutritional interventions, quantitative findings and applicability conditions to their source documents. Two-stage rule-based cleaning first repaired structural noise; domain-focused compression then reorganised whole publications into knowledge units for LightRAG indexing. We compared baseline and LightRAG-enhanced responses within DeepSeek v4, Qwen 3.7 Max and GLM 5.3 on 100 frozen questions. ChatGPT and Kimi-k3 independently evaluated the same question–response records under a common blinded rubric. Figure 1 summarises the compression workflow and paired evaluation.

Results

Domain corpus construction

Data collection, PDF conversion, rule-based cleaning and knowledge compression yielded a corpus covering nutrient requirements, ration formulation and DMI. Other topics included rumen fermentation, milk production, transition-period management, feed efficiency and health risks.

The source collection comprised 3,549 literature files, including 3,518 PDFs, and 966 structured feed records. The feed records comprised 682 entries in the final feed database and 284 in the National Academies of Sciences, Engineering, and Medicine (NASEM) feed library. Deduplication, relevance screening, full-text completeness checks and formatting quality control yielded 2,925 Markdown files representing eligible documents. These comprised 858 Chinese publications, 2,019 English publications and 48 standards.

PDF conversion yielded 2,925 documents, with 2,423 processed using PyMuPDF4LLM and 502 supported by Marker. Conversion logs recorded 0 failures, 2 low-text files and 0 files flagged for manual review.

Two-stage rule-based cleaning reduced text volume from 119,017,051 to 112,532,801 non-whitespace characters, corresponding to a 5.45% reduction and 94.55% retention.

Following knowledge compression and supplementation with standards text, the final corpus contained 2,925 documents and 6,992,080 non-whitespace characters. Before supplementation, compression

reduced the cleaned literature text from 112,532,801 to 5,025,354 characters, corresponding to a 95.53% reduction and 4.47% retention. The net increase between the compressed literature and final corpus was 1,966,726 characters. Supplementary Table S1 summarises the stages of corpus construction and their reported inventories.

Cleaning and knowledge compression

Rule-based cleaning and knowledge compression performed complementary roles. Cleaning repaired document structure, whereas compression selected feeding-relevant evidence and organised it around study subjects, conditions, measurements and source identifiers (Supplementary Table S2). This separation allowed content reduction to follow structural repair without treating formatting removal as domain knowledge selection.

A representative case illustrates the removal of page separators and isolated page numbers, correction of abnormal Chinese spacing and repair of some fragmented terms. Text structure and domain numerical values were retained in the example. Supplementary Fig. S1 presents an illustrative reconstruction of the progression from source content through cleaned text to selected knowledge units.

References and tables were retained by default because substantive text may follow chapter-level reference lists. Tables can also contain diet composition, nutrient measurements, treatment groups and production outcomes. This conservative approach reduced the risk of erroneous deletion but could retain low-value content requiring further control during compression and sample review.

Compression organised publication content into knowledge units linking study subjects and physiological stages to interventions, measurements, findings, management implications and sources. The extraction instructions prioritised numerical values, units, statistical qualifiers and applicability conditions. Changes to these elements or to causal relationships were defined as compression errors requiring source review. Supplementary Fig. S1 illustrates this representation.

Test questions and response records

The evaluation used 100 questions and six response configurations: three base models, each with and without LightRAG. The 600 response records were evaluated by ChatGPT and Kimi-k3, yielding 1,200 total-score observations. Question identifiers and question text matched exactly across the two scoring datasets. Each response was paired with the corresponding baseline or LightRAG response from the same question and base model.

Each judge received one question and one response without reference answers or competing responses to that question. Both judges used the same prompts, four-component rubric and blinding conditions, as confirmed by the researchers. The original response table contained one DeepSeek v4 LightRAG no-context failure message and 599 other response records. All 600 records were retained; the no-context

record received 0 from ChatGPT and 7.40 from Kimi-k3. The two judges therefore contributed 300 within-model comparisons each, rather than additional independent response-generation trials.

LightRAG knowledge base

Construction records reported that all 2,925 imported documents completed parsing and indexing without import failures, corresponding to a document-processing anomaly rate of 0%. Text splitting yielded 5,610 chunks, with a mean length of 1,246.36 characters and a median of 1,445 characters. LightRAG extracted 52,509 entities and 64,863 relations, with a mean node degree of 2.4705. Entity types included concepts, data, methods, artefacts, organisms, natural objects, content, events, people, places and organisations. Concepts and data were the largest categories, comprising 17,566 and 15,993 entities, respectively. Highly connected entities included precision feeding, milk yield, dairy cows, rumen fermentation, DMI, concentrate-to-forage ratio, rumen pH, feed efficiency and milk production. These connections formed an initial network linking feeding, nutritional regulation, rumen function and production. Supplementary Fig. S2 shows a local graph view, and Fig. 2 summarises corpus and knowledge-base statistics.

Question–answer matching under two independent judges

ChatGPT scores increased with LightRAG for all three base models: DeepSeek v4, 55.78 to 78.67 (mean paired gain 22.89; 95% CI 20.76–24.57); Qwen 3.7 Max, 43.54 to 79.16 (mean paired gain 35.62; 95% CI 34.52–36.68); GLM 5.3, 65.26 to 80.18 (mean paired gain 14.92; 95% CI 13.79–16.08). Kimi-k3 scores increased with LightRAG for all three base models: DeepSeek v4, 39.76 to 80.63 (mean paired gain 40.87; 95% CI 38.29–43.21); Qwen 3.7 Max, 31.58 to 80.30 (mean paired gain 48.72; 95% CI 47.07–50.33); GLM 5.3, 44.71 to 82.77 (mean paired gain 38.06; 95% CI 36.39–39.76). Holm-adjusted P values were 7.34×10^{-17} for ChatGPT–DeepSeek and 2.24×10^{-17} for each of the other five comparisons (two-sided asymptotic Wilcoxon tests). Figure 3 and Table 1 report paired effects, including the no-context failure record.

Table 1
Judge-specific paired question–answer matching with and without LightRAG.

Judge	Base model	Baseline	LightRAG	Δ [95% CI]	W/T/L
ChatGPT	DeepSeek v4	55.78	78.67	22.89 [20.76–24.57]	99/0/1
ChatGPT	Qwen 3.7 Max	43.54	79.16	35.62 [34.52–36.68]	100/0/0
ChatGPT	GLM 5.3	65.26	80.18	14.92 [13.79–16.08]	100/0/0
Kimi-k3	DeepSeek v4	39.76	80.63	40.87 [38.29–43.21]	98/0/2
Kimi-k3	Qwen 3.7 Max	31.58	80.30	48.72 [47.07–50.33]	100/0/0
Kimi-k3	GLM 5.3	44.71	82.77	38.06 [36.39–39.76]	100/0/0

All scores range from 0 to 100. Each row summarises 100 question-matched pairs within one base model and one judge. Δ denotes mean LightRAG minus baseline; brackets show 95% paired percentile-bootstrap confidence intervals (20,000 question resamples). W/T/L denotes counts of positive/zero/negative differences. All 600 response records were retained for each judge, including the no-context DeepSeek record (ChatGPT 0; Kimi-k3 7.40). All six judge-specific comparisons had Holm-adjusted $P < 0.001$. Confidence intervals are marginal, not multiplicity-adjusted.

Within-model paired comparisons and judge agreement

ChatGPT assigned higher LightRAG scores in 299 of 300 pairs: wins/ties/losses were 99/0/1 for DeepSeek v4 and 100/0/0 for both Qwen 3.7 Max and GLM 5.3. Kimi-k3 favoured LightRAG in 298 of 300 pairs, with corresponding counts of 98/0/2, 100/0/0 and 100/0/0. The judges agreed on the direction of the difference in 299 of 300 pairs (99.67%); the sole directional disagreement occurred for DeepSeek question 74. A secondary equal-weight average of the two judges gave mean paired gains of 31.88, 42.17 and 26.49 points, respectively.

Across all 600 records, Pearson r was 0.895 (95% CI 0.88–0.91) and absolute-agreement ICC(A,1) was 0.764 (95% CI 0.74–0.78). Agreement depended on response configuration: baseline-only Pearson r and ICC(A,1) were 0.635 and 0.259, compared with 0.554 and 0.528 for LightRAG responses. Kimi-k3 minus ChatGPT mean differences were − 16.18 points for baseline responses and + 1.90 points for LightRAG responses. Thus, the judges supported the same model-level retrieval conclusion while differing in score calibration and question-level rankings (Fig. 4; Supplementary Table S6).

Discussion

Document-level knowledge compression provided a domain-oriented interface between heterogeneous publications and graph/vector indexing. Treating a publication as a coherent study unit was intended to

keep subjects, interventions, quantitative findings and applicability conditions together while removing less relevant background. The reported compression stage retained 4.47% of the cleaned literature text.

The two judges consistently favoured the LightRAG-enhanced configuration at the model level, while assigning different absolute scores. Kimi-k3 scored baseline responses lower on average, making its estimated improvements larger. Reporting each judge separately therefore gives a clearer account of robustness than averaging scores alone. The agreement in paired direction supports the retrieval comparison under this rubric.

The evaluation measured question–answer matching, coverage, clarity and boundary awareness on a fixed domain question set. The original responses and ChatGPT scores underwent record-level human review; the additional Kimi-k3 totals were checked for complete question alignment and valid score ranges. These checks confirmed completeness, question alignment and arithmetic consistency of the supplied scoring records.

This study combined domain-focused document-level compression with LightRAG to make a multisource dairy-feeding corpus available for question answering. Its contribution lies in organising evidence around study conditions, quantitative findings and source links, together with evaluation across three base models and two independent judges. Both judges found positive mean LightRAG gains for all three models, and their question-level gain directions agreed in 299 of 300 pairs. The results support the workflow for domain question answering under the tested conditions and provide a basis for subsequent compression-specific and expert validation.

Methods

Data sources and corpus construction

We combined manual searches and programmatic collection to construct a corpus for dairy cow precision feeding. We manually searched and screened Chinese literature through China National Knowledge Infrastructure. Search terms corresponded to dairy cow precision feeding, dairy cow feeding and dairy feed. English literature was collected through open scholarly search interfaces, covering dairy cow precision feeding, ruminant nutrition models and total mixed ration (TMR) optimisation. The search also covered transition-cow nutrition, rumen fermentation and smart dairy farms. National, industry and local standards were checked and archived manually.

Programmatic collection recorded titles, authors, publication years, unique identifiers, open-access status and full-text links. Records were deduplicated using DOIs, database identifiers and normalised titles. Records were excluded when full text could not be obtained lawfully, topical relevance was insufficient, the main text was substantially incomplete or files were corrupted. For multiple versions of the same work, we prioritised complete versions with clear provenance.

The corpus also included feed ingredient and nutrient composition information from professional resources such as Feedipedia[12], together with standards relevant to dairy management. Structured fields included dry matter, crude protein (CP), neutral detergent fibre (NDF), acid detergent fibre (ADF), starch, energy, calcium and phosphorus. File names, source URLs and retrieval dates were retained to support traceability.

PDF conversion and text preprocessing

Each publication was processed as a unit during PDF-to-Markdown conversion for subsequent cleaning, compression and indexing. Source file paths and processing logs were retained to link knowledge entries to the original evidence.

PDFs with regular layouts and predominantly continuous text were processed in batches using PyMuPDF4LLM[13]. The conversion aimed to preserve heading hierarchies, paragraphs, lists and identifiable tables. Conversion logs recorded successful conversions, failures and files with low text content. Resume support reduced duplicate processing and omissions during batch execution.

Marker (marker-pdf, version 1.10.1) [14] supported conversion of complex publications containing equations, multipage tables, multiple columns or mixed text and graphics. Manual spot checks assessed paragraph order and the preservation of equations, tables and units. Marker complemented the primary PyMuPDF4LLM (version 0.0.27) [13] workflow and was applied selectively to complex layouts, rather than to all publications.

Preprocessing normalised encoding, line breaks and heading levels, and repaired fragmented paragraphs, broken English words, abnormal Chinese spacing and encoding errors. Subsequent rule-based cleaning addressed page numbers, headers, footers, contents pages, copyright statements, image placeholders and repeated headings. Tables, equations, legends and references directly relevant to dairy feeding were retained in principle to reduce domain knowledge loss.

The resulting raw Markdown corpus was accompanied by file-level logs of conversion tools, processing times, character counts, anomaly types and manual-review status. This corpus served as the input to two-stage rule-based cleaning.

Rule-based cleaning and quality control

Raw Markdown often contained page separators, residual contents entries, cover metadata, headers, footers, abnormal line breaks, incorrect heading hierarchies and duplicate files. We applied basic rule-based cleaning followed by postprocessing, retaining inputs, outputs and summary logs at each stage.

The first stage normalised characters and structure by standardising line breaks and whitespace and removing invisible characters and obvious page noise. It also repaired abnormal Chinese spacing and English line-break hyphenation, normalised headings and merged abnormal line breaks. Conversion failures and files with unusually little text were flagged during this stage.

The second stage addressed residual document-level artefacts through title correction, repeated-title handling and Markdown structure normalisation. It removed cover and contents remnants, recurring headers and footers, and high-frequency short lines. It also repaired merged abstracts and body text and applied custom optical character recognition corrections.

Cleaning was conservative and did not remove references or tables by default. Dissertations may contain chapter-specific reference lists, and tables may contain diet composition, nutrient measurements, treatment groups and production outcomes. Material was removed only when it was clearly invalid, severely disordered or redundant with the main text.

Each cleaning round recorded titles, source file names and paths, applied rules, character counts before and after cleaning, failure status and anomaly types. Duplicate documents were identified using normalised text hashes and checked against titles, publication years and source information.

The two stages aimed to standardise heading hierarchies, text continuity and formatting consistency across the Markdown corpus.

Knowledge compression and corpus optimisation

Full publications contain background, repeated explanations and procedural detail that can occupy retrieval context without directly answering feeding questions. We applied document-level knowledge compression after rule-based cleaning to select feeding-relevant evidence and reorganise it into a structured representation. The processing unit was the publication, so that experimental subjects, treatments and outcomes could be considered together before downstream chunking and graph extraction.

The compression strategy imposed domain-specific content requirements. Outputs prioritised study subjects and physiological stages; feeding or nutritional interventions and treatment conditions; and measurements including DMI, CP, NDF, ADF, net energy for lactation (NEL), rumen fermentation, milk production and composition, and health indicators. The prompts specified numerical values, units, statistical significance, applicability boundaries and original document identifiers as required fields or qualifiers when present in the source. Here, compression denotes selection and reorganisation of text, rather than parameter distillation.

Literature was compressed using DeepSeek V4 Flash through an API-based workflow, with temperature 0.0 and a maximum output length of 5,000 tokens. No additional top-p parameter or explicit input truncation threshold was specified. Prompts prohibited adding information absent from the source and instructed the model to preserve numerical values, units, P values, treatment groups, variable names, microbial names and metabolite names. Outputs were structured by study subjects and physiological stages, nutritional interventions, key measurements, principal findings and management implications, with original source-file identifiers. These instructions specified preservation objectives rather than a measured guarantee of lossless compression. Failed API requests were retried up to three times, and persistent failures were logged.

The workflow used whole-document compression to maintain the study context before indexing. Documents exceeding the model context limit were logged separately and processed with structure-aware segmentation. Engineering observations motivated this choice because chunk-level processing could separate subjects, treatment conditions and findings across sections or tables. Those observations were not treated as a controlled comparison of compression strategies. Compression reduction was calculated as 100 × (1 − compressed characters / cleaned characters), using non-whitespace character totals from the same reported processing stage. Final-corpus supplementation was reported separately from literature compression.

LightRAG knowledge-base construction and question answering

The compressed corpus was imported into a Docker deployment of LightRAG [7] (version 1.4.16) for indexing. The inspected deployment used token-based chunking implemented in tiktoken, with a maximum chunk size of 1,200 tokens and an overlap of 100 tokens. During index construction, Qwen/Qwen3-32B [15] was used to extract entities and relations. DeepSeek v4, Qwen 3.7 Max and GLM 5.3 separately generated final responses in their respective evaluation configurations. We used BAAI/bge-m3 [16] to generate the 1,024-dimensional embeddings for indexing. Entities were merged by exact matching of normalised names, and entity types were selected by frequency. Descriptions were deduplicated and concatenated or summarised by the language model. Relations sharing the same unordered pair of endpoints were merged by accumulating weights and combining keywords and descriptions. Source chunk identifiers were retained subject to configured limits during this merging process. Archived query requests used mix mode to combine graph-based retrieval with direct vector retrieval of text chunks. Retrieval parameters were top_k = 20 for entities and relations and chunk_top_k = 8 for text chunks. Context budgets were 3,000 tokens for entities, 4,000 tokens for relations and 12,000 tokens overall.

The knowledge base supported dairy nutrition and feeding-management questions by retrieving text chunks, entities and relations as context for each base model. Source chunk and document identifiers maintained links from indexed evidence to the underlying publications. The availability of those links was distinct from an answer-level audit of the correctness and completeness of citations.

Evaluation design

Test questions

The evaluation dataset comprised 100 questions on dairy cow precision feeding, generated according to researcher-defined objectives and domain-specific requirements. The questions covered nutritional requirements, ration formulation, dry matter intake regulation, total mixed ration management and lactation nutrition. They also addressed transition-period metabolic health, rumen fermentation, feed ingredient evaluation, milk composition, health risks and precision feeding decision support. The dataset included 25 factual, 27 explanatory, 30 consultative and 18 comprehensive decision-making questions.

Reference answers and supporting evidence were subsequently compiled from dairy cow nutrition references, veterinary resources and agricultural extension materials.

All evaluation configurations used the same 100 questions, which were not revised in response to system outputs during scoring. Retrospective within-set screening identified no exact duplicates or question pairs exceeding a normalised character-level similarity threshold of 0.80. A SHA-256 checksum was generated for the currently available question dataset to support subsequent integrity verification.

Paired comparisons evaluated the addition of LightRAG within each base model. DeepSeek v4, Qwen 3.7 Max and GLM 5.3 each had baseline and LightRAG-enhanced configurations. Each question contributed one response pair per model, yielding 300 question-level pairs and 600 response records. Between-model results were descriptive; differences across providers were not interpreted as LightRAG effects.

The question set and model groups were frozen before collecting the baseline and LightRAG-enhanced responses. The two judges then evaluated the same response records independently. The original ChatGPT component scores and review records were retained; Kimi-k3 supplied an additional total score for each record. All comparisons were reconstructed from question identifiers, exact question text and model/configuration labels, without regenerating or editing the answers.

Comparators and experimental settings

Each base model was evaluated with and without LightRAG using the same questions. Analysis focused on question-level paired differences, reported separately for DeepSeek v4, Qwen 3.7 Max and GLM 5.3. This design distinguished within-model retrieval comparisons from between-provider score differences (Supplementary Table S3).

ChatGPT and Kimi-k3 each received one question and its corresponding response, with generating-model identity and baseline or LightRAG-enhanced status concealed. The researchers confirmed that the same scoring prompts, rubric and blinding conditions were used for both judges. Reference answers, historical scores, external model descriptions and competing responses to the same question were excluded.

All 600 response records were retained in both judge analyses. The no-context record was not replaced or excluded. Scoring-data completeness was defined by the presence of a numerical score for every question–model–configuration combination, separately from whether the underlying answer cell contained text.

Evaluation measures and statistical analysis

Both judges used a 100-point rubric comprising question matching (0–35), coverage (0–30), clarity (0–20) and boundary awareness (0–15). Each score assessed one question and its corresponding response (Supplementary Tables S4 and S5).

Question matching assessed whether responses addressed the subject, terminology and task. Coverage assessed whether responses included definitions, mechanisms, differences, measures or other requested elements. Clarity assessed structure, readability and technical expression in each response. Boundary awareness assessed conditional qualifications, cautious wording and warnings about high-risk numerical values or recommendations. Automated review flags identified very short responses, low question relevance, insufficient coverage of multipart questions, unconditional numerical statements and excessive length or redundancy. Supplementary Table S4 summarises the rubric used to score these four components.

ChatGPT and Kimi-k3 evaluated records independently without reference answers or scores from the other judge. Response length was not an independent rubric component. The original ChatGPT response and score records had undergone individual researcher review. Additional checks verified that all 100 question identifiers and question strings matched exactly, all 1,200 total scores were present and within 0–100, and the 600 original component-score sums equalled their corresponding totals.

The primary summaries retained the two judges separately. For each judge and base model, we calculated the 100 question-matched differences as LightRAG minus baseline, their mean, and wins, ties and losses. Mean-difference 95% confidence intervals used a paired percentile bootstrap with 20,000 resamples of the 100 questions and random seed 20261104. The question was the resampling unit, preserving baseline–LightRAG pairing. For a secondary descriptive summary, the two judge scores were averaged within each response before calculating paired differences; the judges were not counted as independent experimental replicates.

Two-sided Wilcoxon signed-rank tests used the asymptotic method, discarded zero differences and applied no continuity correction. Holm adjustment was performed over the six judge-specific model contrasts; the secondary averaged-score comparisons formed a separate family. Reported bootstrap intervals are marginal intervals and were not adjusted for multiplicity. All observations, including the no-context record and both stored scores for it, were retained.

Judge agreement was characterised using Pearson correlation, Spearman correlation, the two-way mixed-effects absolute-agreement intraclass correlation coefficient for single scores, ICC(A,1), and the mean Kimi-k3 minus ChatGPT difference. We examined all 600 scored records as well as the baseline and LightRAG subsets to distinguish pooled separation of configurations from agreement within a configuration. Confidence intervals for agreement statistics used 3,000 question-cluster bootstrap resamples, retaining all six response configurations within each sampled question. The proportion of matching signs of the 300 within-model score differences measured agreement in the direction of the retrieval comparison.

The paired comparisons estimate the effect of adding LightRAG to each tested base model under the recorded generation and scoring conditions. Between-model rankings and pooled correlations were interpreted descriptively.

The expanded analysis used the existing ChatGPT scores and newly supplied Kimi-k3 totals. Human review of the original records was distinct from judge agreement, and data-integrity checks were distinct from expert source verification. Anonymisation, shared rubric and independent inputs were used to limit information available to each judge.

Qualitative case analysis was not used as a quantitative endpoint in this expanded scoring analysis. The corpus-processing example in Supplementary Fig. S1 illustrates the representation of population, intervention, measurements and source information; it is not a blinded expert adjudication of generated answers.

The analysis comprised 100 questions, 600 response records, 300 within-model pairs per judge and 1,200 judge-specific total-score observations. The scoring scale comprised question matching (35 points), coverage (30), clarity (20) and boundary awareness (15). Analyses used Python 3.12.14, NumPy 2.3.5, SciPy 1.18.0, pandas 3.0.1 and openpyxl 3.1.5.

Declarations

Acknowledgements and funding

This research was funded by the Major Science and Technology Support Program of Hebei Province, grant number 252N7403D.

Competing Interests

Leifeng Guo is a member of the Editorial Board of Scientific Reports and a Guest Editor of the “Precision Livestock Farming and on-Farm AI” Collection. He is also the academic supervisor of Bowen Li. The authors declare no other competing interests.

Author Contribution

Bowen Li developed and implemented the experimental workflow and wrote the manuscript. Yongsheng Si, Chendong Yang, Hongtian Li, Paul Kwan, Leifeng Guo and Yabin Ma supervised the research and reviewed the manuscript.

Code availability

The code package associated with this study is publicly available on GitHub at https://github.com/l2933734202-jpg/-. The package includes scripts for corpus collection, cleaning and

compression, LightRAG-based question answering, paired statistical analysis and figure generation, together with dependency information and usage instructions.

Data Availability

The evaluation dataset, comprising 100 test questions, 600 response records and the corresponding evaluation scores, together with the processed data used for paired statistical analysis, is publicly available at https://github.com/l2933734202-jpg/-. The score workbooks and analysis tables are provided in the inputs and analysis folders within the deposited archive. Corpus-processing and knowledge-base statistics are reported in the article and its Supplementary Information. Full-text source publications are not redistributed in the repository and should be accessed through their original publishers or databases under the applicable access conditions.

References

1. 1. National Academies of Sciences, Engineering, and Medicine. *Nutrient Requirements of Dairy Cattle: Eighth Revised Edition* (The National Academies Press, 2021). https://doi.org/10.17226/25806.
2. 2. Ji, Z. et al. Survey of hallucination in natural language generation. *ACM Comput. Surv.* **55**, 248 (2023). https://doi.org/10.1145/3571730.
3. 3. Lewis, P. et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems* **33**, 9459–9474 (2020). https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.
4. 4. Ji, S., Pan, S., Cambria, E., Marttinen, P. & Yu, P. S. A survey on knowledge graphs: representation, acquisition, and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **33**, 494–514 (2022). https://doi.org/10.1109/TNNLS.2021.3070843.
5. 5. Pan, S. et al. Unifying large language models and knowledge graphs: a roadmap. *IEEE Trans. Knowl. Data Eng.* **36**, 3580–3599 (2024). https://doi.org/10.1109/TKDE.2024.3352100.
6. 6. Edge, D. et al. From local to global: a Graph RAG approach to query-focused summarization. Preprint at https://arxiv.org/abs/2404.16130 (2024).
7. 7. Guo, Z., Xia, L., Yu, Y., Ao, T. & Huang, C. LightRAG: simple and fast retrieval-augmented generation. Preprint at https://arxiv.org/abs/2410.05779 (2024).
8. 8. Es, S., James, J., Espinosa Anke, L. & Schockaert, S. RAGAs: automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations* 150–158 (Association for Computational Linguistics, 2024). https://doi.org/10.18653/v1/2024.eacl-demo.16.
9. 9. Silva, B., Nunes, L., Estevão, R., Aski, V. & Chandra, R. GPT-4 as an agronomist assistant? Answering agriculture exams using large language models. Preprint at

https://arxiv.org/abs/2310.06225 (2023).

10. 10. Yang, S., Liu, Z. & Mayer, W. ShizishanGPT: an agricultural large language model integrating tools and resources. Preprint at https://arxiv.org/abs/2409.13537 (2024).
11. 11. Zhang, Z., Wilson, C.-A., Hay, R., Everingham, Y. & Naseem, U. BeefBot: harnessing advanced LLM and RAG techniques for providing scientific and technology solutions to beef producers. In *Proceedings of the 31st International Conference on Computational Linguistics: System Demonstrations* 54–62 (Association for Computational Linguistics, 2025). https://aclanthology.org/2025.coling-demos.7/.
12. 12. INRAE, CIRAD, AFZ & FAO. Feedipedia: animal feed resources information system. https://www.feedipedia.org/ (accessed 21 September 2026).
13. 13. Artifex Software, Inc. PyMuPDF4LLM. https://pymupdf.readthedocs.io/en/latest/pymupdf4llm/ (accessed 21 September 2026).
14. 14. Datalab. Marker. https://github.com/datalab-to/marker (accessed 21 September 2026).
15. 15. Yang, A. et al. Qwen3 technical report. Preprint at https://arxiv.org/abs/2505.09388 (2025).
16. 16. Chen, J. et al. M3-Embedding: multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024* 2318–2335 (Association for Computational Linguistics, 2024). https://doi.org/10.18653/v1/2024.findings-acl.137.

Figures

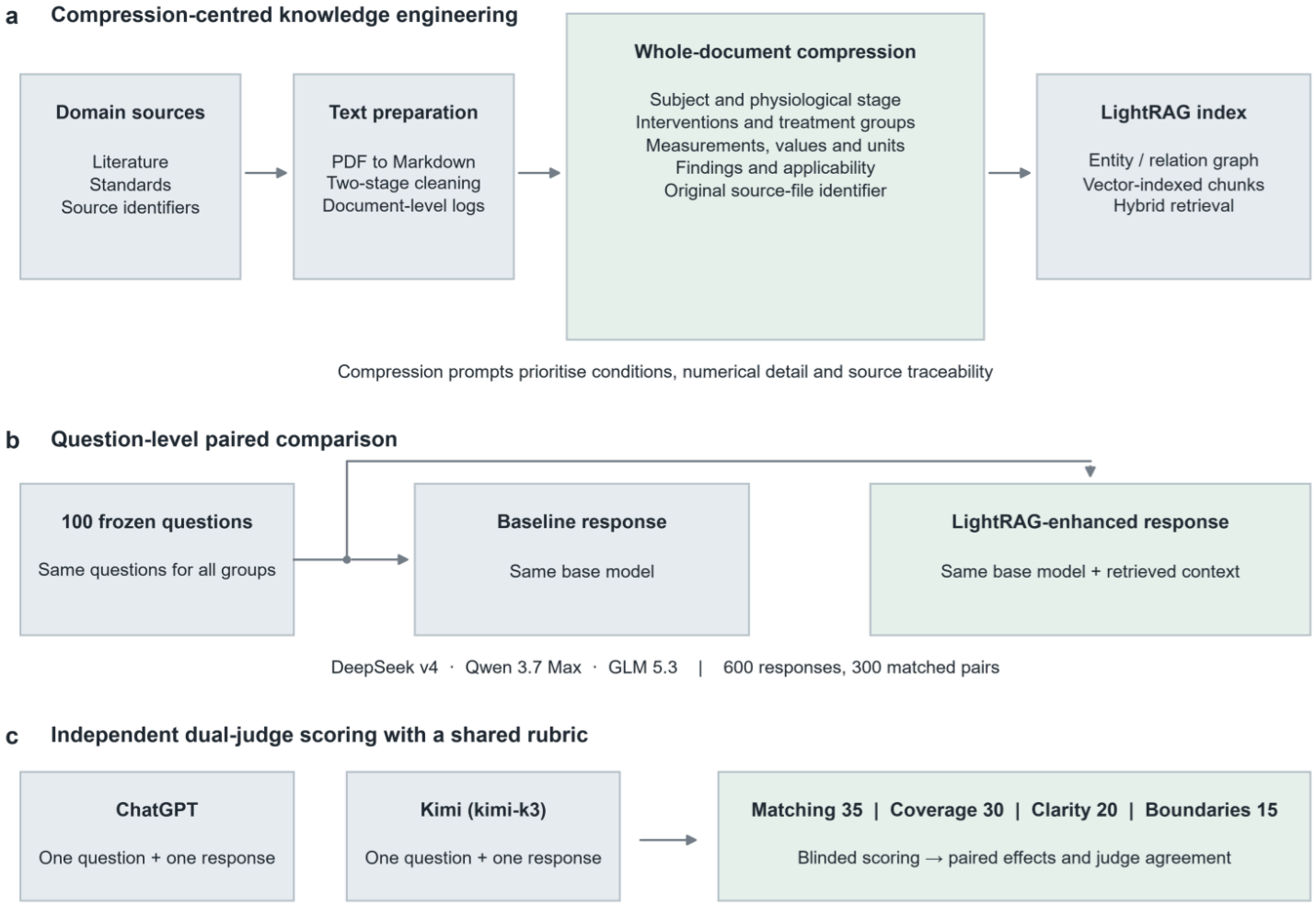


Figure 1

Compression-centred knowledge engineering and dual-judge paired evaluation for dairy cow precision-feeding question answering. a, Domain sources undergo PDF-to-Markdown conversion and two-stage rule-based cleaning before document-level knowledge compression and LightRAG graph/vector indexing. Compression prompts prioritise subjects and physiological stages, interventions and treatment groups, measurements and numerical units, findings and applicability, and original source-file identifiers. These are extraction requirements rather than an empirical measure of information preservation. b, The same 100 frozen questions independently enter baseline and LightRAG-enhanced generation branches using each of three base models: DeepSeek v4, Qwen 3.7 Max and GLM 5.3. The 600 responses form 300 pairs matched by question and base model. c, ChatGPT and Kimi (API identifier kimi-k3) independently score each question–response record under an identical blinded protocol and 100-point rubric: question matching, 35; coverage, 30; clarity, 20; and boundary awareness, 15. Paired effects and judge agreement are analysed after scoring.

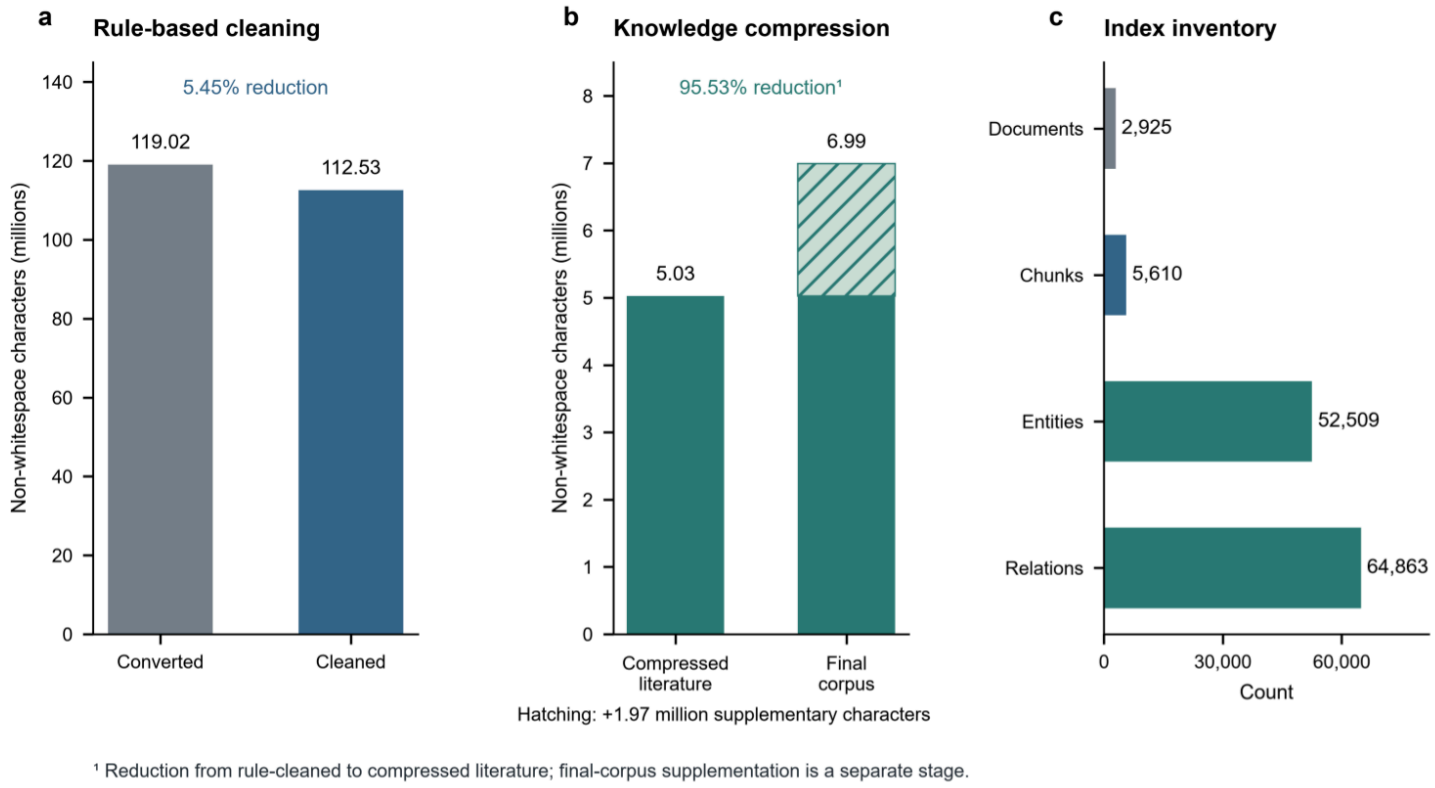


Figure 2

Corpus-processing volumes and the LightRAG index inventory. a, Two-stage rule-based cleaning reduces the manuscript-reported text inventory from 119,017,051 to 112,532,801 non-whitespace characters, a 5.45% reduction. b, Document-level knowledge compression reduces the cleaned literature to 5,025,354 characters, a 95.53% reduction relative to the cleaned-literature stage. Subsequent supplementation increases the final corpus to 6,992,080 characters; hatching marks the additional 1,966,726 characters. Compression and supplementation therefore use distinct stage definitions. c, The manuscript-reported index contains 2,925 documents, 5,610 chunks, 52,509 entities and 64,863 relations. Source data are supplied in the accompanying corpus-stage and index-count CSV files.

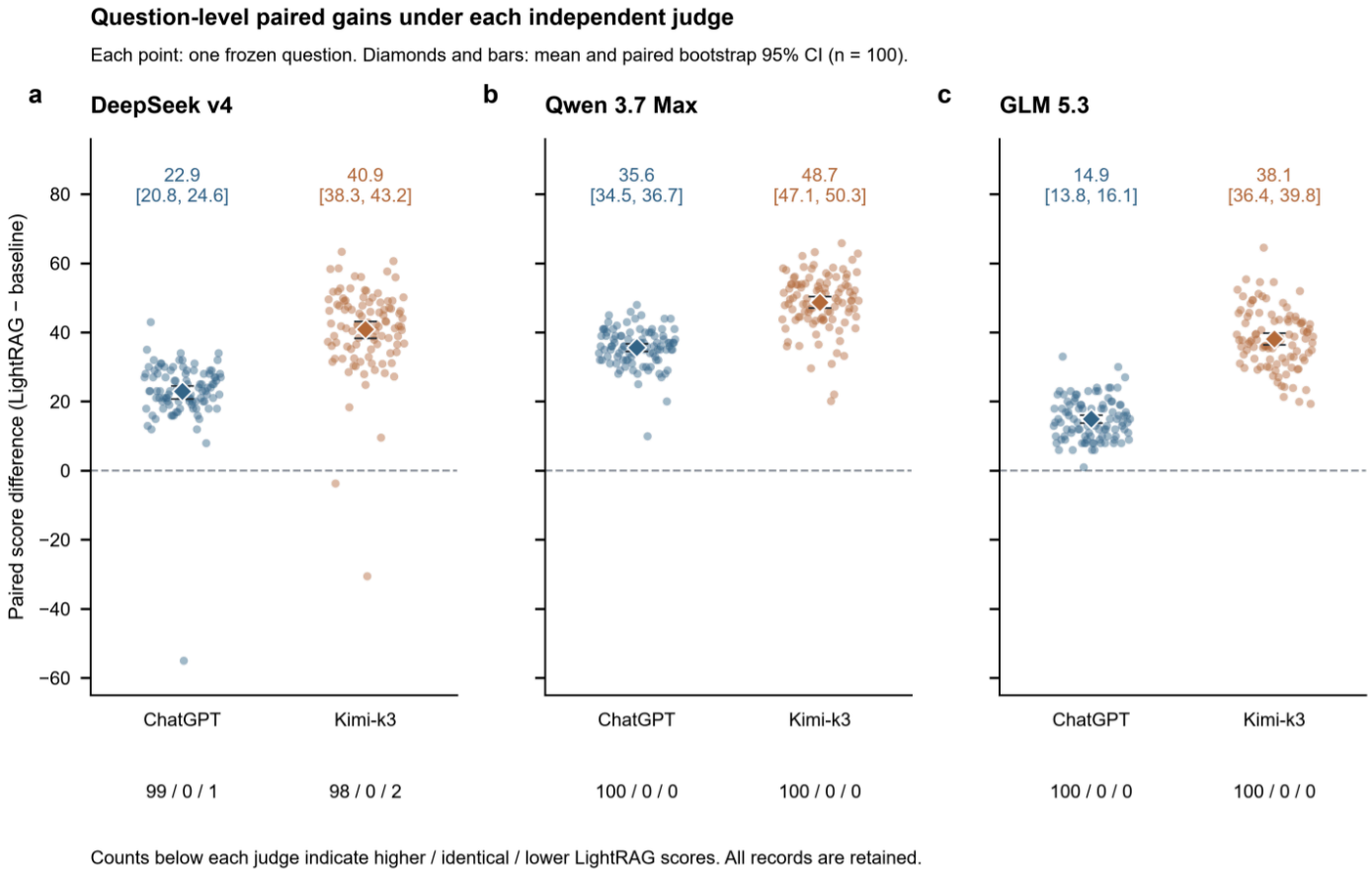


Figure 3

Question-level paired score differences under two independent judges. a–c, LightRAG-enhanced minus baseline question–answer matching scores for DeepSeek v4, Qwen 3.7 Max and GLM 5.3, respectively. Blue points denote ChatGPT and orange points Kimi-k3. Each point represents one of the same 100 frozen questions within a base model and judge; horizontal jitter separates overlapping points. Diamonds show mean paired differences and error bars show percentile 95% confidence intervals from 20,000 paired question-level bootstrap resamples. Numerical annotations report the mean and interval. Counts below each judge indicate higher, identical and lower LightRAG scores. All 600 scored responses per judge are included. The DeepSeek v4 LightRAG no-context failure at question 56 is retained at its original scores of 0 (ChatGPT) and 7.40 (Kimi-k3). Scores quantify question–answer matching under the shared rubric. Source data are supplied in the paired-score and paired-summary CSV files.

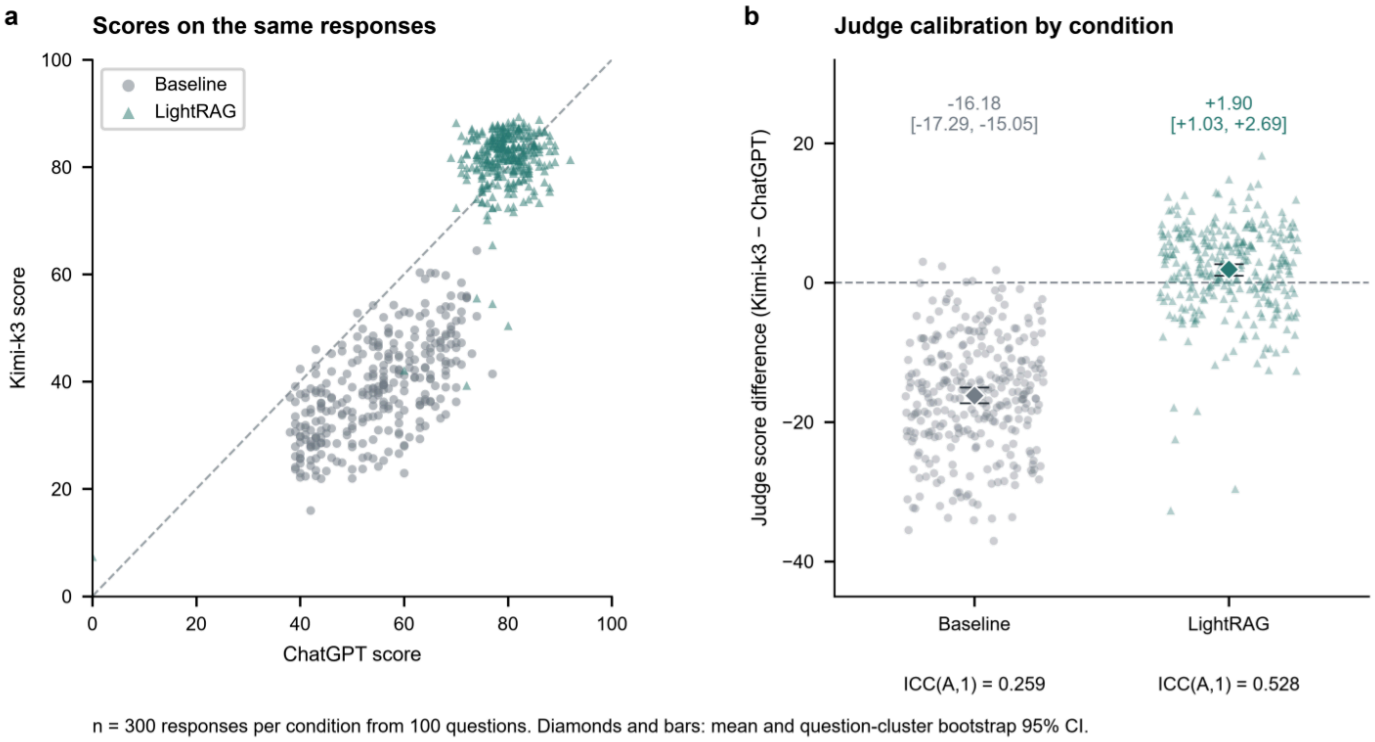


Figure 4

Judge-score association and condition-dependent calibration. a, ChatGPT and Kimi-k3 scores for the same 600 responses; grey circles denote baseline responses and teal triangles LightRAG-enhanced responses. The dashed diagonal denotes identical scores. b, Kimi-k3 minus ChatGPT scores for 300 responses in each condition, derived from the same 100 questions and three base models. Diamonds show mean judge differences and bars show 95% confidence intervals from question-cluster bootstrap resampling, preserving all model records from each selected question. Labels report the mean and interval; lower annotations report the two-way absolute-agreement single-score intraclass correlation coefficient, ICC(A,1). The dashed horizontal line denotes zero judge difference. Kimi-k3 assigns lower baseline scores on average, while the two judges' mean LightRAG scores are closer. Score association and agreement do not establish expert-adjudicated factual correctness. Source data are supplied in the response-level judge-agreement and agreement-summary CSV files.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInformation.docx