Construction of a Japanese RAG Benchmark Using Synthetic Documents on Non-existent Entities and Events

Shengzhe Li[1,2] **Masaya Ohagi**[1] **Hayato Tsukagoshi**[1] **Akihiko Fukuchi**[1]
Tomohide Shibata[1] **Daisuke Kawahara**[2]
[1] SB Intuitions Corp. [2] Waseda University
shengzhe.li@{sbintuitions.co.jp, asagi.waseda.jp}, dkw@waseda.jp
{akihiko.fukuchi, tomohide.shibata}@sbintuitions.co.jp

Abstract

Retrieval-augmented generation (RAG) is a technique in which a large language model (LLM) generates answers based on relevant documents retrieved from an external document collection. Existing RAG evaluation benchmarks often use public data, such as Wikipedia and news articles, as the external document collection. However, these data are highly likely to be already included in the LLM's pre-training corpus, which may prevent an accurate evaluation of the model's ability to generate answers based on the retrieved documents. In this study, we construct a Japanese RAG benchmark by having an LLM synthesize documents about non-existent entities and events and use this collection of synthetic documents as the search target. Since these synthetic documents are not included in the LLM's training data, the ability to generate answers based on retrieved documents can be evaluated more accurately. In addition to the synthetic documents, the benchmark is composed of questions and correct answers, which are created using a combination of LLMs and human effort. We then evaluated and analyzed the RAG performance of existing LLMs using the constructed benchmark.

Keywords: Japanese, RAG, benchmark, synthetic, non-existent

1. Introduction

Retrieval-augmented generation (RAG) is a technique where a large language model (LLM) generates answers based on relevant documents retrieved from an external document collection (Lewis et al., 2020; Gao et al., 2023). RAG is widely applied and has attracted attention because it enables the generation of answers for non-public documents, such as recent events an LLM has not been trained on or confidential internal documents.

However, accurately evaluating this capability remains a significant challenge. Conventional RAG evaluation benchmarks predominantly rely on publicly available materials, such as Wikipedia or news articles (e.g., Saad-Falcon et al. (2023); Yu et al. (2024)). Consequently, it is highly likely that the documents targeted for retrieval are already included in the LLM's pretraining corpus. This raises a severe concern regarding **data contamination**: if the documents have already been learned, the evaluation may fail to accurately assess the model's reliance on retrieved context (Krishna et al., 2024).

In fact, when evaluating RAG with existing benchmarks, there are cases where an LLM can answer correctly without referring the relevant documents. In the case shown in Fig. 1, we confirmed that the model could answer correctly without being provided the relevant documents. Such cases become an evaluation of the LLM's internal knowledge and fail to assess its ability to generate answers based on retrieved documents. Furthermore, it is expected that future LLMs will acquire even more knowledge. As a result, existing benchmarks based on subjects like Wikipedia and past events will become more solvable using solely an LLM's internal knowledge, and it is anticipated that evaluating the ability to generate answers based on retrieved documents will become even more difficult.

[Relevant Documents]
- iPod（アイポッド）は、Appleが開発・販売する携帯型デジタル音楽プレイヤー。... (iPod is a portable digital music player developed and sold by Apple....)
- Apple Inc.（アップル）は、カリフォルニア州クパチーノに本社を置くアメリカ合衆国の多国籍テクノロジー企業である。... (Apple Inc. is an American multinational technology company headquartered in Cupertino, California....)

[Question] iPodを製作している企業の本社所在地は？(What is the headquarters location of the company that makes the iPod?)
[Correct Answer] カリフォルニア州クパチーノ(Cupertino, California)
[Responses by GPT-4o]
With referring to relevant documents カリフォルニア州クパチーノ(Cupertino, California)
Without referring to relevant documents カリフォルニア州クパチーノ(Cupertino, California)

Figure 1: An example of a case where a model can answer correctly without referencing the relevant documents when JEMHopQA (Ishii et al., 2023) is used for RAG evaluation.

To accurately evaluate RAG's ability to generate answers based on retrieved documents, we construct a Japanese RAG benchmark using a collection of LLM-generated synthetic documents as the retrieval target, instead of web documents that may be included in an LLM's pretraining corpus. We aim to demonstrate the feasibility of our methodology in a middle-resource language, where LLMs are capable of fluent generation. In this study, we use

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 7435–7445
11–16 May 2026.

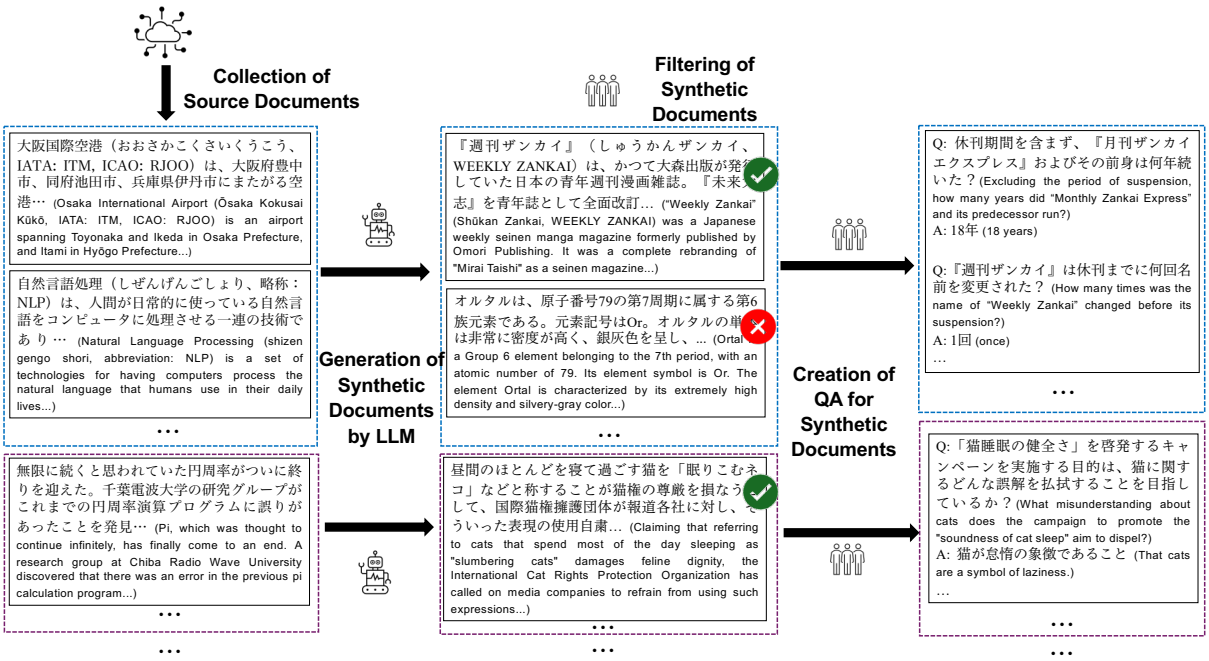

Figure 2: Overview of the benchmark construction procedure

Japanese, a typical middle-resource language, as our testbed. Specifically, we construct the benchmark by having an LLM generate documents about **non-existent entities and events**, and then creating questions and their corresponding correct answers (QA pairs) for that document collection. Since these synthetic documents are not included in an LLM's pretraining corpus, they help mitigate the problem of data contamination during evaluation, enabling a more accurate assessment of the ability to generate answers based on retrieved documents. Note that synthetic documents containing descriptions that contradict real-world facts are manually removed. This is to prevent discrepancies in the evaluation setup between LLMs that possess knowledge of those real-world facts and those that do not.

The construction procedure is shown in Fig. 2. First, we collect several types of base documents and have an LLM create documents about non-existent entities and events that imitate them. Then, those that contradict real-world facts are manually removed. For the remaining documents, we create QA pairs using a combination of an LLM and human effort.

In summary, our contributions are threefold:

1. We describe a RAG benchmark construction methodology designed to mitigate data contamination by utilizing LLMs to synthesize documents about non-existent entities and events, followed by rigorous human filtering.

2. We construct and release[1] a Japanese RAG benchmark comprising 161 documents and 561 QA pairs across four practical domains. The creation of this dataset verifies the feasibility and applicability of our methodology in a middle-resource language.

3. Using the constructed benchmark, we conduct an evaluation of several publicly available LLMs and analyzed their ability to extract information from retrieved documents and to reason using that extracted information.

The remainder of this paper is organized as follows. §2 reviews related work, focusing on existing RAG benchmarks and the challenge of data contamination. §3 outlines the proposed benchmark construction procedure, including the synthesis of documents about non-existent entities and the creation of corresponding QA pairs. §4 describes the experimental setup and discusses the evaluation results of current LLMs. Finally, §5 concludes the work and §6 outlines limitations.

2. Related Work

RAG Benchmarks With the widespread application of RAG (Lewis et al., 2020; Gao et al., 2023), evaluating its performance has become a critical research area. Conventional RAG evaluation benchmarks (Yu et al., 2024) predominantly rely on publicly available materials. For instance, many benchmarks utilize Wikipedia (Saad-Falcon et al., 2023;

[1]To prevent the benchmark from being inadvertently scraped and included in the pretraining corpora of future LLMs, the dataset is released as a password-protected ZIP file at https://github.com/sbintuitions/nonexistent_japanese_rag_benchmark.

Es et al., 2023) or news articles (Chen et al., 2024; Tang and Yang, 2024; Lyu et al., 2024) as the external document collection. Other works focus on domain-specific documents acquired through web crawling or manual collection (Friel et al., 2024; Nishiwaki et al., 2024).

For Japanese, a typical middle-resource language, existing resources include the Allganize RAG Leaderboard (Lee et al., 2024), which evaluates models based on business documents. Additionally, general-purpose QA datasets like JEMHopQA (Ishii et al., 2023), originally constructed from Japanese Wikipedia, have been adapted to evaluate Japanese RAG systems (Yano et al., 2024).

Data Contamination in RAG Evaluation While existing benchmarks provide valuable insights, they face a critical challenge: data contamination. Because many of these benchmarks use public web data, it is highly likely that the documents targeted for retrieval are already included in the LLMs' massive pretraining corpora. In RAG evaluation, if the generator LLM has already memorized the target documents, there is a significant risk that the evaluation cannot be conducted accurately (Krishna et al., 2024; Park et al., 2025).

In such cases, an LLM might bypass the retrieved context and correctly answer questions relying solely on its internal parametric knowledge (as demonstrated in Fig. 1). This conflates the model's memorization capabilities with its actual ability to ground generated answers in the provided retrieved documents. Furthermore, as future LLMs acquire even more knowledge, existing benchmarks based on Wikipedia and past events will become increasingly solvable without retrieval. To address this fundamental issue, our work moves away from public corpora and explores the use of synthetic documents concerning non-existent entities, ensuring that the target information is strictly absent from any pretraining data.

3. Benchmark Construction

This section outlines the benchmark constructed in this study, details its construction procedure, and presents an analysis of the data it contains.

3.1. Benchmark Overview

In this study, we construct a RAG benchmark where the information extracted from retrieved documents (hereinafter called "external information") is essential for every question. To achieve this, we set documents concerning non-existent entities and events—which are unlikely to be in an LLM's pretraining corpus—as the retrieval target within the benchmark. This benchmark is composed of a collection of such documents, in addition to corresponding QA pairs.

It is not practical to manually create all of the documents concerning non-existent entities and events. Therefore, we employ LLMs to synthesize documents about non-existent entities and events based on documents collected from the Web. Specifically, we collect four types of documents commonly used in RAG applications: encyclopedias, news articles, product reviews, and internal company regulations. Then, we prompt LLMs to imitate each of these document types to create documents about non-existent entities and events. Additionally, to prevent evaluation impacts arising from conflicts with LLMs' internal knowledge, only documents that do not contradict real-world facts are targeted for QA pair creation. We manually verify whether the LLM-synthesized documents meet the aforementioned requirements and remove any that do not.

In creating the QA pairs, we aim to measure the ability to utilize external information, which consists of *the ability to extract information from given retrieved documents* and *the ability to reason using the extracted information*. First, to evaluate the former one alone, we create extractive-type (EXTRACTING) QA pairs, where the correct answer to a question is described almost verbatim within the relevant documents. This allows us to evaluate whether an LLM can appropriately extract the parts of the text corresponding to the question when given the retrieved documents. Next, as an evaluation that also includes the latter ability, we create more advanced QA pairs where simply extracting information from the relevant parts of a document is not sufficient to derive the correct answer, and further reasoning using that information is required. This makes it possible not only to evaluate the ability to extract information from the given retrieved documents but also to measure the ability to reason using that extracted information.

Next, we describe the four types of datasets constructed for this benchmark in §3.2, explain the construction procedure for the synthetic document collection in §3.3.1, §3.3.2, and §3.3.3, and detail the procedure for QA construction in §3.3.4.

3.2. The Four Types of Datasets

In this benchmark, we construct the following four types of datasets, envisioning various scenarios where RAG would be necessary. We specifically selected these four domains because they comprehensively represent the most prevalent real-world applications of RAG systems: open-domain knowledge querying (`Wikipedia`), temporal and dynamic event tracking (`News`), customer support and e-commerce (`ProductReview`), and enterprise-level private data retrieval (`CompanyRules`).

『週刊ザンカイ』（しゅうかんザンカイ、WEEKLY ZANKAI）は、かつて大森出版が発行していた日本の青年週刊漫画雑誌。『未来大志』を青年誌として全面改訂する形で1993年に月2回刊誌の『ザンカイ』として誕生した。... 誌面の独特なデザインや、予告ページでのイラストと文が秀逸であることでも知られていた。(“Weekly Zankai” (Shūkan Zankai, WEEKLY ZANKAI) was a Japanese weekly seinen manga magazine formerly published by Omori Publishing. It was launched in 1993 as the semimonthly magazine “Zankai”, a complete rebranding of “Mirai Taishi” as a seinen magazine. ... It was also known for its unique page design and the excellence of the illustrations and text in its preview pages.)

Figure 3: Example of a Pseudo Wikipedia (`Wikipedia`) document.

農業用IoT機器を手がけるクアドラ技研は、次世代の収穫システムとして「米袋ドローン」を発売する計画を発表した。空を飛んで田んぼから直接米袋を回収し、農家まで自動で運搬するとして、農業関係者から熱い注目が集まっている。(Quadra Giken, a company specializing in agricultural IoT devices, has announced plans to release the “Rice Bag Drone” as a next-generation harvesting system. The drone is gathering keen interest from the agricultural community for its ability to fly over rice paddies, directly collect bags of rice, and automatically transport them to farmers.)
...
発売は来年春を予定しており、価格は1機あたり約200万円と少々値が張るものの、国からの補助金制度も検討されており、農家の負担を軽減する見通しも示された。「未来の農業が、空から見えてきた」と農業関係者の間では大きな期待が広がっている。(The launch is scheduled for next spring, and while the price is somewhat steep at around 2 million yen per unit, a government subsidy program is being considered, which is expected to reduce the financial burden on farmers. Great anticipation is spreading throughout the agricultural community, with one member saying, “The future of agriculture has become visible from the sky.”)

Figure 4: Example of a Pseudo News (`News`) document.

「HeatBloom Cardigan」で寒い季節もスタイリッシュに(Stay Stylish Even in the Cold Season with the “HeatBloom Cardigan”)

公開日：2025年1月3日(Publication Date: Jan. 3, 2025)

冬場のカーディガンは暖かさ重視で選びたい反面、ビジネスシーンなどでスマートに見せたい人も多いはず。「HeatBloom Cardigan」はそんな要望を叶えてくれる、新時代の一着です。(While many want to choose a winter cardigan with a focus on warmth, there are also many who want to look smart in business settings. The “HeatBloom Cardigan” is a new-era garment that grants both wishes.)

...(rest omitted)

Figure 5: Example of a Pseudo Product Review (`ProductReview`) document.

1. **Pseudo Wikipedia** (`Wikipedia`): A dataset composed of introductions and explanations of non-existent entities, synthesized by imitating articles about real entities found on Wikipedia. While it adopts a style and structure similar to actual encyclopedia articles, its content is fictional. It is designed to simulate newly published encyclopedia articles. An example of a synthesized document is shown in Fig. 3.
2. **Pseudo News** (`News`): A dataset composed of news articles reporting on fictional events, synthesized by imitating articles from a fictional news media outlet. It is intended to measure the ability to handle new chronological information by simulating articles that report on the latest, daily-updated events. An example of a synthesized document is shown in Fig. 4.

X株式会社賃金規程(# X Corporation Wage Regulations)

第1章総則(## Chapter 1: General Provisions)

（目的(Purpose)）

第1条この規程は、就業規則（以下「規則」という。）第38条に基づいて、社員の賃金に関する事項を定めたものである。(Article 1: These regulations are established to define matters concerning employee wages, based on Article 38 of the Work Rules (hereafter referred to as "the Rules").)

（適用範囲(Scope of Application)）

...(rest omitted)

Figure 6: Example of a Pseudo Company Rules (`CompanyRules`) document.

3. **Pseudo Product Review** (`ProductReview`): A dataset consisting of descriptions of specifications and user experiences for non-existent products, synthesized by imitating product evaluation articles found on review sites and personal blogs. This is designed with use cases in mind where RAG is applied to chatbots for product-related inquiries. An example of a synthesized document is shown in Fig. 5.
4. **Pseudo Company Rules** (`CompanyRules`): A dataset consisting of documents describing work systems, compensation structures, and employee benefits, modeled after real-world internal regulations and work rules. The documents have a tone and structure characteristic of internal corporate documents. It assumes a scenario where an internal bot handles information from non-public regulations. An example of a synthesized document is shown in Fig. 6.

3.3. Details of the Construction Procedure

In this section, we describe the construction procedure for the benchmark built in this study. We first collected documents that served as the basis of the synthetic documents, and prompted LLMs to synthesize documents by imitating the collected base documents. As the documents in the benchmark should handle non-existent entities and events, the synthesized documents were manually filtered because they could potentially be about real entities or events. Finally, based on the resulting documents about non-existent entities and events, we created QA pairs that require referencing information within those documents to answer. These processes are described in detail below.

3.3.1. Collection of Base Documents

In this study, we aim to construct a benchmark using documents about non-existent entities and events. However, creating such documents manually would incur a significant cost and is not practical. Therefore, we employ LLMs to synthesize these docu-

ments. To synthesize documents suitable for a RAG benchmark, we prepare a set of topics that envision RAG application scenarios and their corresponding document collections in advance. Then, for each topic, we provide LLMs with real-world documents (hereinafter called "base documents") and instruct LLMs to imitate them, thereby synthesizing documents about non-existent entities and events.

First, we collected the base documents corresponding to each dataset. We established four topics for the base documents: encyclopedias, latest information, product reviews, and internal corporate documents. These represent important use cases for the practical application of RAG.

1. `Wikipedia`: We extracted the first paragraph of Japanese Wikipedia articles as entity definition sentences (Onoe et al., 2022) and collected these texts, which correspond to the opening of an encyclopedia article.
2. `News`: We collected 1,290 articles from the web media outlet "Kyoko Shimbun"[2] (Fictional News)[3].
3. `ProductReview`: We manually accessed and saved product introduction pages from multiple genres on the price comparison site "Kakaku.com"[4], and obtained the text by removing HTML tags.
4. `CompanyRules`: We manually organized documents from the "Model Collection of HR and Labor Regulations"[5] and gathered 24 documents pertaining to regulations.

3.3.2. Synthesizing Documents with LLM

Based on the collected base documents, we synthesized documents concerning non-existent entities and events using an LLM. Specifically, for `Wikipedia` and `News`, we synthesized one imitative document with an LLM for each base document. On the other hand, since the number of base documents for `ProductReview` was relatively small, we synthesized documents using multiple LLMs for each base document to improve diversity. For `CompanyRules`, we created the data under the assumption that all 24 internal regulations belonged to one specific company. During this process, to prevent contradictions among the synthesized regulations, each time a new regulation was generated, we checked for inconsistencies with the previously created ones using both an LLM and manual review. If a contradiction was found, the document was either corrected or regenerated. Through this, we created a mutually consistent set of regulations for a single company.

For the synthesis of `Wikipedia` and `News`, we used OpenAI's `GPT-4o-20240806` to generate non-existent encyclopedia and news articles, providing them with various styles and content. The prompt used for document synthesis was created with reference to Chang et al. (2024). For the synthesis of `ProductReview`, we used OpenAI's `o1-preview-2024-09-12` and `o1-mini-2024-09-12`, and Google's `gemini-1.5-pro`. For the synthesis of `CompanyRules`, we used Google's `gemini-1.5-pro`.

> オルタルは、原子番号79の第7周期に属する第6族元素である。元素記号はOr。オルタルの単体は非常に密度が高く、銀灰色を呈し、... (Ortal is a Group 6 element belonging to the 7th period, with an atomic number of 79. Its element symbol is Or. The element Ortal is characterized by its extremely high density, a silvery-gray color, ...)

Figure 7: An example of a document removed due to contradiction with real-world facts.

3.3.3. Removal of Synthetic Documents That Do Not Meet the Criteria

We asked annotators to judge whether the documents synthesized in §3.3.2 could be used for the benchmark. Specifically, we defined three requirements: (1) **It deals with non-existent entities or events**, (2) **It does not contradict facts**, and (3) It does not contain logical failures, excessive inclusion of information unrelated to the main topic, or obvious deficiencies in anaphoric relations. To rigorously enforce these requirements, annotators were explicitly instructed to conduct web searches for any specific proper nouns (e.g., names of people, organizations, or products) and events mentioned in the generated texts. If a search engine query returned a match indicating that the entity or event actually existed in the real world, the document was immediately discarded. We removed all documents that failed to meet any of these requirements. An example of a document removed for failing to meet condition (2) is shown in Fig. 7: the element with atomic number 79 is actually gold (Au), and a discrepancy in the evaluation setup would arise between an LLM that learned this real-world fact during pre-training and one that did not. Specifically, the former would need to *replace an already learned fact with non-existent information*, so it would be considered to be solving a different task than the latter, for which this is not necessary. To prevent such discrepancies in the evaluation setup arising from the presence or absence of an LLM's

[2] https://kyoko-np.net/

[3] Initially, we attempted to generate synthetic documents by collecting real news articles in the same manner, but the generated articles tended to mix fact and fiction, making it difficult to determine if a synthetic article was truly about a non-existent entity or event. Therefore, in this study, we ultimately judged it was more manageable to use articles from "Kyoko Shimbun" as the base documents and adopted this policy.

[4] https://kakaku.com/

[5] https://www.ohno-jimusho.co.jp/download/

internal knowledge, we removed documents that contradicted real-world facts. The remaining documents were evaluated as having natural Japanese expression and meeting the requirements of the benchmark. We proceeded with the creation of QA pairs based on these documents.

3.3.4. Creation of QA Pairs for the Synthetic Documents

The creation of QA pairs in this benchmark was fundamentally performed by annotators. The process involved two stages: the creation of QA pairs and their filtering. First, we provided annotators with the documents from each dataset (`Wikipedia`, `News`, `ProductReview`, `CompanyRules`) and instructed them to create two types of QA pairs: simple ones (EXTRACTING) that can be answered merely by extracting information from the document, and advanced ones that require further reasoning based on the extracted information. The advanced QA pairs were further subdivided into three reasoning types often required in RAG applications: (1) questions requiring logical reasoning to derive an answer (REASONING), (2) questions requiring calculation (COMPUTING), and (3) questions requiring the integration and organization of information within the document (INFORMATION INTEGRATING). Including the extractive type, the definitions and examples of the four question types established in this study are as follows.

1. EXTRACTING: The correct answer to the question is described almost verbatim within the document, and the model derives the answer by extraction. An example is shown in Fig. 8(a)[6].
2. REASONING: The model derives the answer using common sense or relatively simple logical reasoning for information not explicitly stated in the document. An example is shown in Fig. 8(b): it is first necessary to extract the information that the "Rice Bag Drone" operates on solar panels, and then combine that with the common sense "there is no sunlight at night".
3. COMPUTING: The model derives the answer by processing and calculating numerical values, times, etc., from the document. An example is shown in Fig. 8(c): it is necessary to extract the information that "one drone can carry a maximum of 30kg of rice" and then perform the calculation "150/30=5".
4. INFORMATION INTEGRATING: The model derives the answer by organizing information from multiple locations within a document or a document collection. This requires not just partial extraction, but also summarization and understanding the intent. An example is shown in Fig. 8(d): it is necessary to aggregate information regarding the magazine's name change history, from its launch to suspension.

In the creation of advanced QA pairs, we did not specify which reasoning pattern to use; annotators created questions using one or more of the three reasoning patterns. Additionally, for

(a) An example of an EXTRACTING type QA pair[6]. The corresponding relevant document is Fig. 4.

[Retrieved Document Excerpt] ...クアドラ技研の広報担当者は、これまで農家が収穫後に重労働を強いられていた米袋の運搬作業が、このドローンによって一気に効率化すると話す。「空を飛ぶことで、交通の混雑や道路の起伏による影響を受けず、短時間で安全に目的地まで届けることができるのが最大の特長です」と自信を見せた。... (... A spokesperson for Quadra Giken stated that the drone will dramatically increase the efficiency of transporting rice bags, a task that has traditionally been strenuous post-harvest labor for farmers. "Its greatest feature is that by flying, it can safely reach its destination in a short time, unaffected by traffic congestion or uneven roads", they said with confidence. ...)

[Question] 米袋ドローンには、農家を収穫後に重労働から解放できる以外に、何のメリットがある？(Besides freeing farmers from heavy labor post-harvest, what other merits does the Rice Bag Drone have?)
[Correct Answer] 空を飛ぶことで、交通の混雑や道路の起伏による影響を受けず、短時間で安全に目的地まで届けられること(That by flying, it is unaffected by traffic congestion or uneven roads and can safely reach its destination in a short time.)

(b) An example of a REASONING type QA pair[6]. The corresponding relevant document is Fig. 4.

[Retrieved Document Excerpt] ...また、太陽光電池を装備しており、日中の稼働時間を無制限に延ばせることから、エコフレンドリーな製品としても注目されている... (... It is also equipped with solar panels, allowing for unlimited operation time during the day, which has also drawn attention to it as an eco-friendly product. ...)

[Question] 米袋ドローンが、夜に無限に動くことができない理由は？(What is the reason the Rice Bag Drone cannot operate indefinitely at night?)
[Correct Answer] 太陽光電池で稼働するため、夜には太陽光がなく電池が切れるから(Because it operates on solar panels, there is no sunlight at night, and the battery will run out.)

(c) An example of a COMPUTING type QA pair[6]. The corresponding relevant document is Fig. 4.

[Retrieved Document Excerpt] ...また、最大で30kgの米を運べるほか、風速15メートルの強風にも耐える設計がなされている... (... Furthermore, it can carry up to 30kg of rice and is designed to withstand strong winds of up to 15 meters per second. ...)

[Question] 150kgの米を一度で輸送するため、米袋ドローンは何台必要？(To transport 150kg of rice at once, how many Rice Bag Drones are needed?)
[Correct Answer] 5台(5 units)

(d) An example of an INFORMATION INTEGRATING type QA pair[6]. The corresponding relevant document is Fig. 3.

[Retrieved Document Excerpt] ...1993年に月2回刊誌の『ザンカイ』として誕生した。2001年に週刊化される際に誌名も『週刊ザンカイ』に変更され、... 2009年に編集方針の転換を余儀なくされ、『週刊ザンカイ』は一時的に休刊となった... (... It was launched in 1993 as the bi-monthly magazine "Zankai". When it became a weekly publication in 2001, its name was changed to "Weekly Zankai", ... In 2009, a change in editorial policy forced "Weekly Zankai" into a temporary suspension ...)

[Question] 『週刊ザンカイ』は休刊までに何回名前を変更された？(How many times was the name of "Weekly Zankai" changed before its suspension?)
[Correct Answer] 1回(Once)

Figure 8: Examples of each question type.

[6]For explanatory purpose, the figures show excerpts from the relevant documents needed to derive the correct answer. However, it should be noted that in the actual evaluation, the correct relevant document is not guaranteed to exist within the search results.

	#Doc	#QA	minL	medL	maxL
Wikipedia	50	165	485	727	1,055
News	47	173	445	785	1,150
ProductReview	40	48	564	1,379	2,838
CompanyRules	24	175	979	1,607	11,478

Table 1: Statistics of the datasets. #Doc, #QA, minL, medL, and maxL indicate the number of documents, number of QA pairs, and the minimum, median, and maximum document lengths (in characters), respectively.

	Wikipedia	News	Product Review	Company Rules
#QA →	165	173	48	175
Extracting	95 (57.6%)	90 (52.0%)	13 (27.1%)	119 (68.0%)
Reasoning	36 (21.8%)	36 (20.8%)	15 (31.2%)	45 (25.7%)
Computing	21 (12.7%)	18 (10.4%)	0 (0.0%)	2 (1.1%)
Information Integrating	21 (12.7%)	31 (17.9%)	24 (50.0%)	11 (6.3%)

Table 2: Number of examples and percentages by question type. Note that because a single question can belong to multiple types, the sum of percentages for each dataset is not necessarily 100%.

ProductReview and CompanyRules, we provided draft QA pairs generated by an LLM to assist the annotators. The QA pairs created in this manner were reviewed by the authors, who classified each pair by type and removed instances that did not fit into the categories described above.

3.4. Dataset Analysis

Dataset Statistics Tab. 1 shows the statistics for each dataset, including the number of documents for retrieval, the number of QA pairs, and the character counts of the retrieval documents. It can be seen that the documents of ProductReview and CompanyRules are relatively long, while those of Wikipedia and News are relatively short.

Distribution of Question Types Tab. 2 shows the statistics of questions in each dataset. The distribution of question types tends to vary by topic: in Wikipedia and News, Extracting questions account for slightly more than half, while Reasoning questions constitute about 1/5. In ProductReview, Information Integrating questions are the most common, while Extracting and Reasoning questions each make up slightly more than 1/4, there are no Computing questions. In CompanyRules, Extracting questions are numerous, and while Reasoning questions make up 1/4 of the total, Computing and Information Integrating questions are relatively scarce.

4. Evaluation Experiment

In this section, using the benchmark constructed in §3, we evaluate the ability of LLMs to utilize external information in RAG and describe the results and our analysis.

4.1. Experimental Setup

Naive RAG Pipeline Naive RAG is a framework where a retriever extracts chunked retrieval-target documents, which are then incorporated into a prompt, and an LLM, acting as the generator, generates the text (Ma et al., 2023). For any given question, the retrieval-target documents comprise all documents within the dataset to which the question belongs. All evaluation results in this paper are based on this Naive RAG pipeline.

Retriever We use dense vector search for retrieval. For the embedding model, we use sbintuitions/sarashina-embedding-v1-1b (SB Intuitions Corp., 2024), which showed high performance on the retrieval task of the Japanese Massive Text Embedding Benchmark (JMTEB) (Li et al., 2024)[7]. Four chunks with the highest cosine similarity to the question are retrieved and incorporated into the prompt during generation.

Chunking We set the maximum chunk length to 400 characters for ProductReview and 200 characters for the other datasets, with a 20-character overlap between chunks.

LLMs We evaluate the following Japanese-capable LLMs.

- OpenAI/GPT-4o-2024-08-06 (GPT-4o)
- tokyotech-llm/Llama-3.1-Swallow-8B-Instruct-v0.3 (Swallow-8B) (Fujii et al., 2024; Okazaki et al., 2024)
- llm-jp/llm-jp-3-{1.8b, 3.7b, 13b}-instruct (llm-jp-{1.8B, 3.7B, 13B}) (Aizawa et al., 2024)
- Qwen/Qwen2.5-{1.5B, 3B, 7B, 14B}-Instruct (Qwen-{1.5B, 3B, 7B, 14B}) (Yang et al., 2024)

Evaluation Procedure In tasks like open-ended question answering, the diversity of model outputs means that simple correctness judgments, such as exact string matching, cannot accurately evaluate the results. Therefore, following the setup of existing leaderboards (Lee et al., 2024), we conducted the evaluation of the generated results using an LLM. This method of automatic evaluation using an LLM is known as LLM-as-a-judge and is known to have a significant correlation with human evaluation (Zheng et al., 2023). In this study, we have an LLM make a binary judgment (match or no match) as to whether a model's response to a question aligns with the correct answer. We used OpenAI's GPT-4o-20240806 as the evaluation LLM. To mitigate fluctuations in the evaluation scores due to the randomness of the judge's output, the LLM-as-a-judge evaluation was conducted three times, and the final judgment on whether the model's answer

[7] Based on preliminary experiments which confirmed that varying the retrieval pipeline did not significantly alter the comparative results or conclusions regarding the generator LLMs' performance, this study does not stress-test alternative retrieval methods or models.

matched the correct one was determined by a majority vote. As the evaluation metric for the RAG system, we used accuracy—based on whether the LLM's response matched the correct answer—to measure if the system could correctly derive the answer via RAG.

4.2. Evaluation Results

Following the experimental setup in §4.1, we conducted a performance evaluation of the LLMs. The evaluation results are shown in Tab. 3. First, in the overall score, which combines the scores from all datasets, GPT-4o demonstrated the highest performance. Additionally, Qwen-14B and Qwen-7B, despite being lightweight, showed high performance, particularly in the `ProductReview` dataset where they performed on par with or better than GPT-4o.

The results for each question type are shown in Tab. 3(b) (EXTRACTING), (c) (REASONING), (d) (COMPUTING), and (e) (INFORMATION INTEGRATING), respectively. First, all LLMs achieved their highest scores among the four question types on the EXTRACTING tasks. This is likely because EXTRACTING measures only the ability to extract information from retrieved documents, whereas the other three types require both extraction and reasoning. For the EXTRACTING type, while GPT-4o showed the best performance overall, on the `ProductReview` dataset, Qwen-14B and Qwen-7B performed on par with or better than GPT-4o. For the other three types, GPT-4o also showed the best overall performance, but for REASONING questions on the `ProductReview` dataset, Qwen-3B performed on par with GPT-4o. Additionally, for the INFORMATION INTEGRATING type, Qwen-7B performed on par with GPT-4o on `News`, while on `ProductReview`, Qwen-7B and Qwen-14B outperformed GPT-4o. For the COMPUTING type, GPT-4o significantly outperformed the other LLMs. The performance of GPT-4o on calculation tasks is known to be high compared to other models (Yang et al., 2024), and it is likely that this result reflects the relative strengths and weaknesses of the LLMs in solving mathematical problems.

4.3. Comparison Based on the Use of Retrieval Results

Since this benchmark is designed with questions that cannot be solved using only an LLM's internal knowledge, it is ideally desirable that it consists exclusively of problems that are unsolvable without the retrieved documents and only become solvable when they are provided. To verify whether this objective has been achieved, we examined how the LLMs' evaluation results change with and without the use of the retrieved documents.

In this verification, for each LLM from the experiments in §4.1, we had them generate answers to

(a)

	Overall				
	Overall (561)	Wikipedia (165)	News (173)	Product Review (48)	Company Rules (175)
GPT-4o	**76.8**	**87.3**	**69.9**	52.1	**80.6**
llm-jp-1.8B	20.7	35.2	25.4	6.2	6.3
llm-jp-3.7B	26.0	30.3	22.5	10.4	29.7
llm-jp-13B	40.5	49.1	37.0	25.0	40.0
Swallow-8B	52.0	61.2	43.4	35.4	56.6
Qwen-1.5B	31.9	40.0	27.7	18.8	32.0
Qwen-3B	36.9	40.0	34.1	37.5	36.6
Qwen-7B	60.6	62.4	55.5	**62.5**	63.4
Qwen-14B	65.6	70.9	58.4	52.1	71.4

(b)

	EXTRACTING				
	Overall (317)	Wikipedia (95)	News (90)	Product Review (13)	Company Rules (119)
GPT-4o	**81.7**	**89.5**	**73.3**	76.9	**82.4**
llm-jp-1.8B	25.2	47.4	25.6	15.4	8.4
llm-jp-3.7B	35.0	44.2	25.6	7.7	37.8
llm-jp-13B	49.5	66.3	43.3	38.5	42.0
Swallow-8B	59.9	71.6	52.2	38.5	58.8
Qwen-1.5B	40.1	50.5	33.3	30.8	37.8
Qwen-3B	43.2	51.6	37.8	46.2	40.3
Qwen-7B	71.6	76.8	65.6	**92.3**	69.7
Qwen-14B	74.8	83.2	66.7	84.6	73.1

(c)

	REASONING				
	Overall (132)	Wikipedia (36)	News (36)	Product Review (15)	Company Rules (45)
GPT-4o	**81.8**	**88.9**	**88.9**	**53.3**	**80.0**
llm-jp-1.8B	18.2	25.0	36.1	6.7	2.2
llm-jp-3.7B	16.7	19.4	25.0	6.7	11.1
llm-jp-13B	34.1	36.1	38.9	13.3	35.6
Swallow-8B	50.8	61.1	47.2	26.7	53.3
Qwen-1.5B	23.5	33.3	25.0	13.3	17.8
Qwen-3B	37.9	33.3	44.4	**53.3**	31.1
Qwen-7B	51.5	50.0	55.6	40.0	53.3
Qwen-14B	64.4	66.7	63.9	40.0	71.1

(d)

	COMPUTING				
	Overall (41)	Wikipedia (21)	News (18)	Product Review (0)	Company Rules (2)
GPT-4o	**70.7**	**81.0**	**66.7**	-	0.0
llm-jp-1.8B	4.9	9.5	0.0	-	0.0
llm-jp-3.7B	4.9	4.8	5.6	-	0.0
llm-jp-13B	9.8	9.5	11.1	-	0.0
Swallow-8B	19.5	23.8	16.7	-	0.0
Qwen-1.5B	12.2	19.0	5.6	-	0.0
Qwen-3B	0.0	0.0	0.0	-	0.0
Qwen-7B	26.8	28.6	27.8	-	0.0
Qwen-14B	46.3	47.6	50.0	-	0.0

(e)

	INFORMATION INTEGRATING				
	Overall (87)	Wikipedia (21)	News (31)	Product Review (24)	Company Rules (11)
GPT-4o	**54.0**	**71.4**	41.9	41.7	**81.8**
llm-jp-1.8B	11.5	9.5	25.8	0.0	0.0
llm-jp-3.7B	12.6	0.0	19.4	12.5	18.2
llm-jp-13B	26.4	19.0	29.0	25.0	36.4
Swallow-8B	33.3	33.3	25.8	37.5	45.5
Qwen-1.5B	19.5	14.3	25.8	12.5	27.3
Qwen-3B	28.7	28.6	29.0	33.3	18.2
Qwen-7B	47.1	33.3	**45.2**	**62.5**	45.5
Qwen-14B	39.1	33.3	32.3	45.8	54.5

Table 3: Complete evaluation results. The values in the tables represent accuracy scores. Part (a) shows the overall results, and parts (b)-(e) show the results for each specific question category. The number in brackets means the number of QAs.

	Overall (561)	Wikipedia (165)	News (173)	Product Review (48)	Company Rules (175)
GPT-4o	14.1	9.1	11.6	29.2	17.1
llm-jp-1.8B	0.4	0.0	0.6	0.0	0.6
llm-jp-3.7B	1.1	0.0	0.0	2.1	2.9
llm-jp-13B	6.4	2.4	1.2	2.1	16.6
Swallow-8B	17.1	7.3	11.6	12.5	33.1
Qwen-1.5B	11.1	4.8	8.1	22.9	16.6
Qwen-3B	14.6	9.1	8.1	22.9	24.0
Qwen-7B	14.8	9.7	9.2	37.5	18.9
Qwen-14B	20.1	14.5	13.9	27.1	29.7

Table 4: Performance of each model when no retrieval results were provided.

the questions without providing the retrieval results composed of relevant documents, and then evaluated those answers. The results are shown in Tab. 4. These results correspond to those shown in Tab. 3 where retrieval results were provided, and the two should be referred to in conjunction. Overall, the accuracy rate without retrieval results was significantly lower compared to the performance with retrieval results. In particular, we confirmed that the performance of GPT-4o—which had consistently performed well in the setting with retrieval results—dropped to a level comparable to that of the other models.

The reasons why the performance shown in Tab. 4 is not zero are threefold. First, the benchmark includes binary-choice questions (e.g., Yes/No), where models can answer correctly by chance. Second, models sometimes generate plausible hallucinations that coincidentally align with the correct answer. Finally, for datasets that mimic real-world formats, such as company regulations, models can leverage pre-trained common-sense knowledge to correctly answer general questions. Despite these artifacts, the accuracy without retrieval is significantly lower across all models than the RAG setting, confirming that the benchmark largely achieves its objective of being unanswerable without the external documents.

4.4. Error Analysis

In the example in Fig. 9, it is necessary to organize the information—"weekly", "No. 1, 2005" (start of the year), and "No. 50, 2012" (end of the year)—before performing the calculation. While GPT-4o is able to answer correctly by taking these requirements into account, Qwen-14B—the model with the second-highest overall performance—is only able to perform a simple calculation, subtracting 2005 from 2012 as written in the document to get 7. Through this example, we can assess that in terms of the ability to utilize external information, GPT-4o has superior performance in calculation, reasoning, and information organization.

[Retrieved Document Excerpt] 『レヴェナント・クロニクル』（Revenant Chronicle）は、井端圭史による日本の漫画作品である。...『週刊クリエイトタイムズ』（晴耕社）の創刊号である2005年1号から2012年50号まで連載され、... ("Revenant Chronicle" is a Japanese manga series by Keishi Ibata. ... It was serialized in "Weekly Create Times" (Seikosha) from the inaugural issue, No. 1 of 2005, to No. 50 of 2012, ...)

[Question] 『レヴェナント・クロニクル』は『週刊クリエイトタイムズ』で約何年間連載された？(For about how many years was "Revenant Chronicle" serialized in "Weekly Create Times"?)
[Correct Answer] 約8年間 (About 8 years)
[Response by GPT-4o] 約8年間 (About 8 years)
[Response by Qwen-14B] 7年間 (7 years)

Figure 9: An example where GPT-4o is correct but Qwen-14B is incorrect (Dataset: Wikipedia)

5. Conclusion

To more accurately evaluate RAG's ability to generate answers based on retrieved documents, this study constructed a RAG benchmark using a collection of LLM-synthesized documents about non-existent entities and events as its retrieval target. The benchmark built in this study includes four types of datasets, each composed of synthetic documents from an LLM and manually created question-answer pairs. Furthermore, using this benchmark, we evaluated the ability to extract external information and to reason using that extracted information within a RAG framework. Through the construction of this benchmark, we have verified that the proposed methodology is applicable to a middle-resource language like Japanese for which LLMs can perform fluent generation. We hope that an accurate evaluation of an LLM's ability to utilize external information, made possible by this benchmark, will serve as a guidepost for developing higher-performance LLMs and RAG systems.

6. Limitations

In this study, we constructed a novel Japanese RAG benchmark using synthetic documents to effectively mitigate data contamination. While our evaluation demonstrates the feasibility of this approach in assessing a model's true ability to utilize retrieved context, our comprehensive analysis also identified several limitations in the current dataset and evaluation setup. In the following, we outline these potential limitations and discuss critical directions for future work to address them.

- Disentangling retrieval and generation evaluation: An inherent limitation involves our evaluation setup. Our current naive RAG pipeline evaluates the system end-to-end, which conflates retrieval errors with generation errors. To precisely pinpoint system bottlenecks, evaluating retrieval accuracy and the LLM's generation capability in isolation – such as testing the generator exclusively with gold documents – is essential. However, distinguishing retrieval and generation

errors needs further annotations specifying the exact source sentences required for each answer, which are not yet included in our current datasets, we leave the separated evaluation of retrieval and generation as a critical direction for future work.

- Scaling the dataset via automated verification: The dataset consists of 161 synthetic documents and 561 QA pairs. This relatively small scale is a direct consequence of the costly construction process, which required LLM generation followed by careful manual filtering and QA creation. To address this scalability issue, we plan to develop an automated, LLM-driven verification pipeline. By utilizing our current human-verified data as a seed, we aim to finetune or prompt an "LLM critic" to reliably detect factual contradictions, significantly reducing human annotation costs and enabling the generation of massive synthetic datasets.

- Improving ecological validity: Our synthetic documents, which are clean and narrative, may not represent the full spectrum of real-world documents, which often include noisy text. Furthermore, while annotators confirmed the fluency of the generated Japanese, it remains unexplored whether these synthetic texts differ from real-world human-authored documents in stylistic or linguistic dimensions (e.g., lexical diversity or syntactic complexity). To enhance the ecological validity of the benchmark, our next step is to introduce controlled noise into the synthetic generation pipeline. It is worth a try to simulate real-world data imperfections by injecting noise such as OCR errors, colloquial disfluencies, and complex structural elements like tables. This may be helpful to bridge the gap between clean synthetic text and practical industrial realities.

- Mitigating evaluation bias: Since the LLM-as-a-judge used in this study also had errors in judging correctness, exploring more robust automated evaluation methods for RAG can be considered another important future task. A related concern is the use of `GPT-4o-20240806` as both the primary judge and as the top-performing model in the evaluation. This coupling could introduce a potential bias, as the judge may favor the response style of its own model. While this choice was made after preliminary tests showed other LLMs to be less reliable judges, and large-scale human evaluation was not feasible, future work could explore cross-judge validation to better quantify and mitigate this bias.

7. Ethical Considerations

An ethical consideration for this benchmark is the dual-use nature of its methodology. The techniques employed to generate "fake but plausible" documents, while benign in this research context, could be adapted for malicious purposes. Specifically, the ability to create realistic-sounding but fictional news, company rules, or encyclopedic entries highlights a potential risk for generating sophisticated disinformation.

8. Bibliographical References

Akiko Aizawa, Eiji Aramaki, Bowen Chen, Fei Cheng, Hiroyuki Deguchi, Rintaro Enomoto, Kazuki Fujii, Kensuke Fukumoto, Takuya Fukushima, Namgi Han, et al. 2024. LLM-jp: A Cross-Organizational Project for the Research and Development of Fully Open Japanese LLMs. *arXiv preprint arXiv:2407.03963*.

Hoyeon Chang, Jinho Park, Seonghyeon Ye, Sohee Yang, Youngkyung Seo, Du-Seong Chang, and Minjoon Seo. 2024. How Do Large Language Models Acquire Factual Knowledge During Pretraining? *arXiv preprint arXiv:2406.11813*.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking Large Language Models in Retrieval-Augmented Generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17754–17762.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. 2023. RAGAS: Automated Evaluation of Retrieval Augmented Generation. *arXiv preprint arXiv:2309.15217*.

Robert Friel, Masha Belyi, and Atindriyo Sanyal. 2024. RAGBench: Explainable Benchmark for Retrieval-Augmented Generation Systems. *arXiv preprint arXiv:2407.11005*.

Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Hiroki Iida, Masanari Ohi, Kakeru Hattori, Hirai Shota, Sakae Mizuki, Rio Yokota, and Naoaki Okazaki. 2024. Continual Pre-Training for Cross-Lingual LLM Adaptation: Enhancing Japanese Language Capabilities. In *First Conference on Language Modeling*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv:2312.10997*.

Ai Ishii, Naoya Inoue, and Satoshi Sekine. 2023. Constructing a Japanese Multi-hop QA Dataset for Evidence-Grounded Explainable Question Answering. In *Proceedings of the 29th Annual Meeting of the Association for Natural Language Processing (NLP2023)*. In Japanese.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. 2024. Fact, Fetch, and Reason: A Unified Evaluation of Retrieval-Augmented Generation. *arXiv preprint arXiv:2409.12941*.

Junghoon Lee, Akiko Oshio, Sounghan Kim, and Yujung Kim. 2024. Allganize RAG Leaderboard. https://huggingface.co/datasets/allganize/RAG-Evaluation-Dataset-JA.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Shengzhe Li, Masaya Ohagi, and Ryokan Ri. 2024. JMTEB: Japanese Massive Text Embedding Benchmark. https://huggingface.co/datasets/sbintuitions/JMTEB.

Yuanjie Lyu, Zhiyu Li, Simin Niu, Feiyu Xiong, Bo Tang, Wenjin Wang, Hao Wu, Huanyong Liu, Tong Xu, and Enhong Chen. 2024. CRUD-RAG: A Comprehensive Chinese Benchmark for Retrieval-Augmented Generation of Large Language Models. *ACM Transactions on Information Systems*.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query Rewriting for Retrieval-Augmented Large Language Models. *arXiv preprint arXiv:2305.14283*.

Gaurav Maheshwari, Dmitry Ivanov, and Kevin El Haddad. 2024. Efficacy of Synthetic Data as a Benchmark. *arXiv preprint arXiv:2409.11968*.

Kazutaka Nishiwaki, Shunsuke Onuma, Tsuyoshi Kudo, and Kazuma Kadowaki. 2024. Performance Evaluation of GPT-4 and RAG for the Automation of Financial Planning. In *Proceedings of the 30th Annual Meeting of the Association for Natural Language Processing (NLP2024)*. In Japanese.

Naoaki Okazaki, Kakeru Hattori, Hirai Shota, Hiroki Iida, Masanari Ohi, Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Rio Yokota, and Sakae Mizuki. 2024. Building a Large Japanese Web Corpus for Large Language Models. In *First Conference on Language Modeling*.

Yasumasa Onoe, Michael JQ Zhang, Eunsol Choi, and Greg Durrett. 2022. Entity Cloze by Date: What LMs Know About Unseen Entities. *arXiv preprint arXiv:2205.02832*.

Chanhee Park, Hyeonseok Moon, Chanjun Park, and Heui-Seok Lim. 2025. MIRAGE: A Metric-Intensive Benchmark for Retrieval-Augmented Generation Evaluation. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 2883–2900.

Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2023. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems. *arXiv preprint arXiv:2311.09476*.

SB Intuitions Corp. 2024. Sarashina-Embedding-v1-1B. https://huggingface.co/sbintuitions/sarashina-embedding-v1-1b.

Yixuan Tang and Yi Yang. 2024. MultiHop-RAG: Benchmarking Retrieval-Augmented Generation for Multi-Hop Queries. *arXiv preprint arXiv:2401.15391*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 Technical Report. *arXiv preprint arXiv:2412.15115*.

Chihiro Yano, Hayato Tsukagoshi, Ryohei Sasano, and Koichi Takeda. 2024. Evaluating Japanese Sentence Embeddings with Document Retrieval and Retrieval-Augmented Generation Tasks. In *Proceedings of the 30th Annual Meeting of the Association for Natural Language Processing (NLP2024)*. In Japanese.

Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. 2024. Evaluation of Retrieval-Augmented Generation: A Survey. *arXiv preprint arXiv:2405.07437*.

Yunpu Zhao, Rui Zhang, Wenyi Li, Di Huang, Jiaming Guo, Shaohui Peng, Yifan Hao, Yuanbo Wen, Xing Hu, Zidong Du, et al. 2024. Assessing and Understanding Creativity in Large Language Models. *arXiv preprint arXiv:2401.12491*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.