國立臺灣大學電機資訊學院暨中央研究院資料科學學位學程

碩士論文

Data Science Degree Program

College of Electrical Engineering and Academia Sinica

National Taiwan University

Master's Thesis

改變校園對話：基於具檢索增強生成的大語言模型聊天機器人

Revolutionizing Campus Conversation: LLM-Powered Chatbot with Retrieval-Augmented Generation

陳姵如

Pei-Ju Chen

指導教授: 陳祝嵩博士、陳駿丞 博士

Advisor: Chu-Song Chen, Ph.D. Jun-Cheng Chen, Ph.D.

中華民國 113 年 8 月

August 2024

doi:10.6342/NTU202400247

Acknowledgements

時光荏苒，回首過去兩年碩士生涯，首先想感謝的是我的指導教授組陳祝嵩與陳駿丞老師，總能敏銳地指出我想法上的疏漏，給我不一樣的想法與觀點以及足夠的運算資源，讓我能順利完成想做的研究。再來想感謝實驗室的夥伴們，廷芸、玉辰、欣玉、Alice、宗維、奕嘉、建維、已畢業的學長們，一起修過的課、研究討論、118 巷覓食的飯後閒聊以及你們的存在都讓我成長很多，看見了更廣闊的世界。最重要的，感謝我的家人，讓我能專心研究無後顧之憂。最後，感謝求學一路上遇到的所有人，很幸運有你們陪我一起走過。

doi:10.6342/NTU202400247

摘要

大型語言模型（LLM）可謂是 2023 年人工智慧領域中最為熱門的名詞之一，而其在各種場景中的應用方法之一即為檢索增強生成（RAG），因此受到廣泛關注。雖然在學術界受到高度重視，但目前在實際生活中的應用仍相對有限。為此，本論文特別選擇了一個與我身為學生緊密相關的場域 ── 學校，成功地將檢索增強生成的大型語言模型應用於實際生活中。在語言模型資源相對較為匱乏的繁體中文情境中，這篇作品整合了多項技術包含網頁爬蟲及網頁資料清理、嵌入檢索（embedding retrieval）、文檔切分最佳化、前後端、line 聊天機器人 UI 整合，最終取得了成功的應用。實現將技術應用到實際情境中的重要里程碑，檢索增強生成的大語言模型朝著實現高品質、個人化支持及普及化的目標邁進。

關鍵字：大語言模型、檢索增強生成、校園個人助理

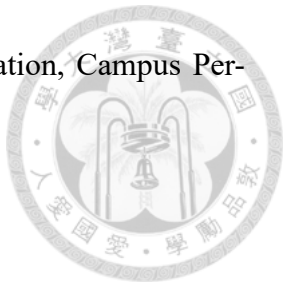
doi:10.6342/NTU202400247

Abstract

Large Language Models (LLMs) can be considered one of the hottest terms in the field of artificial intelligence in 2023, and one of their application methods among various scenarios is Retrieval-Augmented Generation (RAG), which attracts widespread attention. Although highly regarded in the academia, their practical applications in real life are relatively limited. Therefore, this paper specifically chooses a field closely related to me as a student—school—and successfully applied the Retrieval-Augmented Generation of LLMs to real-life situations. In the context of Traditional Chinese where language model resources are relatively scarce, this work integrates multiple technologies, including web crawling and data cleaning, embedding retrieval, chunk optimization, front-end and back-end development, and Line chatbot UI integration, ultimately achieving successful application. This milestone in applying technology to real-world scenarios propels LLMs in Retrieval-Augmented Generation towards the goals of achieving high-quality, personalized support, and widespread use.

doi:10.6342/NTU202400247

Keywords: Large Language Models, Retrieval-Augmented Generation, Campus Personal Assistant



doi:10.6342/NTU202400247

Contents

doi:10.6342/NTU202400247

doi:10.6342/NTU202400247

List of Figures

doi:10.6342/NTU202400247

List of Tables

doi:10.6342/NTU202400247

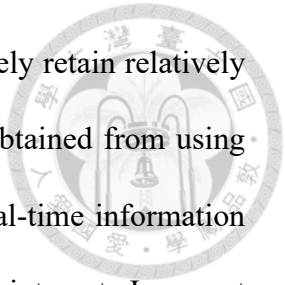
Chapter 1 Introduction

1.1 Motivation

In the contemporary quest for information, individuals now have an additional avenue aside from Google: consulting ChatGPT. However, both approaches come with their own set of limitations. Firstly, Google's search engine results may not consistently lead users to the precise information they seek, necessitating manual exploration of web pages to locate relevant details. On the other hand, while ChatGPT eliminates the need for users to search for information snippets manually, its data is limited to the training set and lacks real-time updates. Furthermore, the reliability of the information provided may be questionable, potentially leading to hallucinations. Recognizing the strengths of both methods, a solution that integrates the merits of Retrieval-Augmented Generation (RAG) has gained substantial attention.

In campus life, we often need to inquire about various administrative issues, such as course selection, registration, academic affairs, and more. These pieces of information are usually scattered across the webpages of different departments and offices, and searching on Google may not immediately yield the required information. We frequently find ourselves filtering through webpages one by one to locate the information we need. Therefore, the university provides an excellent domain for the application of large language models.

doi:10.6342/NTU202400247



However, directly using large language models often fails to effectively retain relatively ”long-tail” and ”time-sensitive” knowledge. The typical response obtained from using large language models is, ”I’m sorry, but I am unable to provide real-time information as my cutoff date is January 2022, and I currently cannot browse the internet. I suggest checking the official website to ensure you receive the most accurate and up-to-date information.” Hence, combining the RAG technique with retrieval methods is well-suited for application in this domain.

In the Retrieval-Augmented Generation(RAG) technology, there exist several approaches. For applications with frequently updated data, our focus lies on the non-training RAG version. This not only eliminates the exorbitant cost associated with creating training datasets but also circumvents the high-cost training phase. Consequently, application development across various industries can proceed without reliance on powerful computational capabilities.

In paper, ICRALM [22], have established superior retrieval models and language models without the need for fine-tuning, whether through joint training or independent training, achieving enhanced results. The current landscape witnesses the flourishing development of large language models, with new and more powerful models emerging every two weeks or a month. It is noteworthy that improved language models often come with larger sizes, higher training costs, and the caveat that many top-ranking models may not be open source, allowing only inference through APIs. Therefore, this work adopts the untrained RAG method to address these challenges.

- In a scenario where only noisy web scraping data is available (with a non-fixed website structure), we have devised an effective data cleaning method and successfully

doi:10.6342/NTU202400247

implemented a RAG (Retrieval-Augmented Generation) system.

- Despite the relatively limited support for traditional Chinese in large language models, we have successfully developed an effective RAG (Retrieval-Augmented Generation) system.
- We have conducted a comprehensive and successful demonstration, showcasing how to obtain annotated data without incurring high manual labeling costs, experimenting with various methods suitable for traditional Chinese usage scenarios, and evaluating a Retrieval-Augmented Generation system.
- We have created a platform for students and faculty at National Taiwan University to assist in addressing common challenges encountered in campus life.

doi:10.6342/NTU202400247

Chapter 2 Related Works

2.1 Retrieval Augmented Generation

The pioneering work that laid the foundation for RAG, known as RAG [17] or REALM [8] originated in 2020. This work introduced the concept of combining retrieval models with generative models. One key difference between past RAG models and the current mainstream models lies in the integration of retrieval and generation models. Previously, RAG models required training to combine the retrieval and generation components. However, the current trend in mainstream RAG models does not necessitate additional training due to the prohibitively high training costs.

Due to the rapid increase in model sizes in the field of LLMs over the past year, the costs associated with fine-tuning LLMs have continued to rise. Moreover, some closed-source LLMs only offer API-based inference, limiting their accessibility. In response to these challenges, the research direction of RAG has shifted towards adopting off-the-shelf LLMs as the mainstream approach. Examples of such developments include [18, 22, 29], among others.

Therefore, the focus of this work is dedicated to the application of off-the-shelf methods, aligning with the current trend in RAG research to make this work more practical.

doi:10.6342/NTU202400247

2.2 Retrieval

2.2.1 Chunk optimization

In the practical implementation of RAG, one encounters the initial challenge of determining the optimal document segmentation for achieving the highest retrieval rate and providing maximum assistance to the language model in generating responses. The current mainstream consensus within the RAG community is that there is no universally established standard; rather, the approach depends on the characteristics of your data and the specific embedding model in use, as highlighted in the [7].

One commonly employed technique in document segmentation is the Sentence-Window Retrieval, or Small-to-Large Chunking, strategy, as discussed in the aforementioned survey [6]. This strategy involves using smaller segments during the retrieval process and utilizing larger, complete paragraphs when feeding the document to the language model for response generation. This approach typically yields better results in both retrieval and response generation stages.

Additionally, recent advancements in the RAG field have introduced a novel document segmentation method, as presented in [4]. This work proposes a method that employs ChatGPT to extract the ”smallest meaningful segments expressing coherent ideas,” referred to as propositions. In the retrieval phase, this approach proves to be more effective than using complete sentences or fixed-length chunk sizes, providing a fresh perspective in addressing this challenge.

doi:10.6342/NTU202400247

2.2.2 Embedding Models

Embedding models can be broadly classified into two main types: Sparse Models and Dense Models. Sparse Models, exemplified by the classic TF-IDF and BM25 [24], employ a dictionary where each document is encoded based on the probability of word occurrences in that document. Given that most words in the dictionary do not appear in a given document, the resulting encoding is predominantly composed of zero values across dimensions, leading to its designation as a ”sparse” model.

On the other hand, models trained using neural networks fall under the category of Dense Models. For instance, DPR [16] utilizes pairs of questions and passages (or answers). Subsequent models often leverage techniques like contrastive learning, as seen in Contriever [13], BGE [33], GTE [19], E5 [31], and others.

Although many of the aforementioned models have demonstrated the ability to handle multiple languages, their training materials predominantly consist of Simplified Chinese documents. Consequently, these models perform exceptionally well in Simplified Chinese and produce reasonably good results in Traditional Chinese. It is anticipated that similarly powerful models developed specifically for Traditional Chinese would exhibit even better performance. However, the development of models tailored to Traditional Chinese faces significant bottlenecks due to the substantially smaller volume of available Traditional Chinese corpora compared to Simplified Chinese.

doi:10.6342/NTU202400247

2.3 Generation

The RAG architecture adopted in our research follows the frozen retrieval model and frozen language model methods, and utilizes the powerful text generation capabilities of LLM as the generator of RAG. Challenges for the generator include understanding the problem and the content of the various documents, figuring out what information the problem requires, and avoiding the influence of irrelevant documents.

State-of-the-art language models such as the GPT series [1, 2] and the LLama series [28], as well as Mixtral-8*7B [14], have approached human expert levels in text understanding. It is anticipated that these models should have no significant issues in comprehending and filtering documents.

However, these language models do not necessarily achieve the same proficiency in Traditional Chinese as in English. Large-scale language models developed in Taiwan include Taide[23], Taiwan-LLama [21] and MediaTek's Breeze-7B [3]. However, the capabilities of the above model are still a considerable challenge under the constraints of limited Traditional Chinese training materials.

2.4 RAG Evaluation

RAG evaluation encompasses two major components, namely the quality of the retrieved documents in the retrieval phase and the quality of the generated text in the generation phase. The evaluation of the retrieval block involves assessing the quality of the retrieved documents in terms of their ability to answer questions and the generation block involves evaluating the quality of the generated text in terms of coherence and the model's

doi:10.6342/NTU202400247

ability to discern document relevance unaffected by irrelevant ones.

In the first part, the retrieval block, traditional metrics such as Recall, Hit Rate, MRR, and NDCG, which often require manual data annotation, are commonly employed. RAGAS[5] proposes a mechanism using LLM as evaluators to reduce manual annotation costs. For the retrieval part, RAGAS introduces the metric Context Relevance, which measures the extent to which the retrieved context contains only the information necessary to answer the question. This is quantified by the proportion of extracted relevant sentences to the total sentences in the context.

In the second part, the generation block, RAGAS[5] introduces the metric Faithfulness to examine whether the answers generated align with the provided documents, and Answer Relevance, which assesses how well the answer addresses the given question.

In addition, another work, ARES [25], introduces an automated evaluation framework that requires only a small amount of manual annotation for a test set to generate a comprehensive dataset for assessing the overall effectiveness of RAG systems across all documents.

doi:10.6342/NTU202400247

Chapter 3 Method

3.1 Problem Definition

The current technological landscape places significant emphasis on the application of LLM. While LLMs have demonstrated remarkable capabilities across diverse benchmarks such as SuperGLUE[30], MMLU [9], and BIG-bench [26] in general scenarios, they exhibit certain limitations when confronted with domain-specific or highly specialized queries [15]. This often results in the generation of incorrect information or ”hallucinations” [36], especially when queries extend beyond the model’s training data or demand up-to-date information.

Therefore, the primary goal of this work is to address the aforementioned weaknesses associated with LLMs in the context of domain-specific tasks. The specific focus is on the application of LLMs within the domain-specific framework of ”National Taiwan University.”

3.2 Database Construction

The primary task in developing a personalized LLM involves collecting domain-specific information. Our approach includes web scraping the official websites of various

doi:10.6342/NTU202400247

departments and units to gather this data. This methodology aligns with our standard information retrieval process for school-related queries, which typically involves searching and browsing through the webpages of different units to obtain relevant information.

Furthermore, the method of web scraping for data collection has the advantage of being ”annotation-free,” allowing for easy adaptation to various application domains.

During web scraping, in addition to extracting textual content, we encounter various multimedia data types such as PDFs and image files (jpg, png, etc.). To handle these diverse data formats, we integrate Optical Character Recognition (OCR) to recognize text within images. This enables us to consolidate the extracted textual information from different media types into a unified text file.

3.3 Test Dataset Construction

To evaluate the RAG system, the constructed test dataset is primarily divided into two main parts: FAQs sourced from web pages and a synthetic dataset generated automatically using ChatGPT.

3.3.1 FAQs sourced from web pages

Due to the absence of a unified structure across the websites of various units at National Taiwan University, the collection of FAQs requires manual searching. We conducted searches on over 200 webpages and ultimately gathered 498 questions. To maintain impartiality, since these FAQs may already exist in our database, we will use ChatGPT to generate similar but not identical questions. The final test data will consist of the questions

doi:10.6342/NTU202400247

generated by ChatGPT along with the standard answers from the FAQ pages.

The following prompt template, shown in Figure 3.1, is provided to ChatGPT for generating questions: We use an zero-shot in-context learning to prompt because we have found that providing examples can sometimes influence the output.

Prompt Template

以下是一組問答：
問： {Question}
答： {Answer}
請基於以上問答生成一個意思一樣的問題給我，不需要回答。
問：

Figure 3.1: Prompt template for ChatGPT question generation.

3.3.2 Synthetic dataset generated using ChatGPT

The primary objective of data synthesis is to evaluate the RAG system's ability to effectively convey information beyond FAQs. Taking inspiration from GenRead [35], which focuses on using GPT-generated answers to retrieve documents rather than dataset creation, its success illustrates that answers generated by GPT can share a certain degree of similarity with documents. Consequently, we feed documents to ChatGPT and instruct it to generate questions and corresponding answers based on the document content, forming our synthesized test dataset with same quantity with FAQ test dataset.

In this process, we employ one-shot in-context learning to prompt ChatGPT as shown in Figure 3.2. This approach enables ChatGPT to grasp the characteristics of the crawled data, distinguish noise, and understand the appropriate methods for posing questions and providing answers.

doi:10.6342/NTU202400247

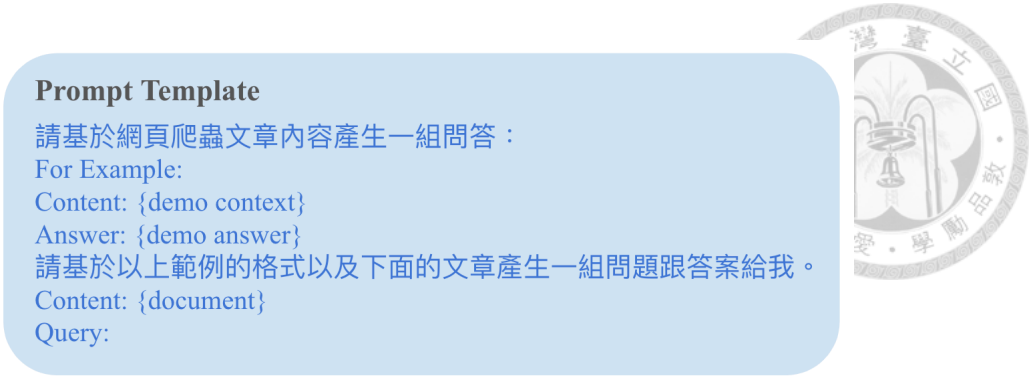


Figure 3.2: Prompt template for synthetic QA generation.

3.4 Retrieval

3.4.1 Chunking and Pre-processing

The reason for combining chunking and pre-processing is that the method we use can achieve both of these effects simultaneously. Inspired by [4], it was found that using ChatGPT to chunk documents, as opposed to segmenting them into sentences or using fixed-length chunks, leads to better retrieval recall.

Moreover, our data characteristics make ChatGPT particularly suitable for document chunking. Not only can it accurately segment documents, but it can also be utilized to filter out noise extracted from web pages. Additionally, it helps rearrange sentences when the webpage structure leads to merging errors in the text paragraphs.

The design of this prompt template primarily draws inspiration from [4], incorporating instructions to eliminate web page noise and including in-context learning examples. Additionally, for more complex tasks, we found that using Chinese instructions alone yielded unsatisfactory results. Based on the insights of [10], we have adopted a mixed Chinese-English prompt, shown in Figure 3.3, approach to achieve optimal chunking results.

doi:10.6342/NTU202400247

Prompt Template

Decompose the "Content" into clear and simple propositions, ensuring they are interpretable out of context.
1. Split compound sentence into simple sentences. Maintain the original phrasing from the input whenever possible.
2. Remove the irrelevant content.
3. For any named entity that is accompanied by additional descriptive information, separate this information into its own distinct proposition.
4. Decontextualize the proposition by adding necessary modifier to nouns or entire sentences and replacing pronouns (e.g., "it", "he", "she", "they", "this", "that") with the full name of the entities they refer to.
5. Present the results as a list of strings, formatted in JSON.
Input: {demo document}
Output:{demo chunks}
Input:"{document}"
Output:

Figure 3.3: Prompt template for synthetic QA generation.

3.4.2 Vectorize

The process of converting prepared chucks into embeddings is referred to as vectorization. We consulted the Huggingface MTEB Leaderboard - Retrieval - Chinese best open-source models and tested the following models: TownsWu/PEG[32], thenlper/gte-large-zh[20], thenlper/gte-base-zh[20], thenlper/gte-small-zh[20], BAAI/bge-large-zh-v1.5[33], BAAI/bge-base-zh-v1.5[33], BAAI/bge-small-zh-v1.5[33], infgrad/stella-large-zh[12], infgrad/stella-base-zh[11], and the Traditional Chinese-trained model yentinglin/bert-base-zhtw[34]. The model identified as BAAI/bge-large-zh-v1.5c [33] demonstrated the best performance and was therefore selected for further experiments. It is noteworthy that the highest-ranked models on the leaderboard may not always be the most suitable for Traditional Chinese applications. Models trained on Simplified Chinese data may outperform those trained exclusively on Traditional Chinese data due to the larger training datasets available for Simplified Chinese.

doi:10.6342/NTU202400247

In application development, apart from accuracy, users are typically sensitive to inference time. Therefore, we selected the best-performing small model, thenlper/gte-small-zh[20], which has an inference time approximately 2.5 times faster than BAAI/bge-large-zh-v1.5[33].

3.5 Generation

In the generation process, we utilize a frozen LLM, incorporating both the query and retrieved information into the prompt, as shown in Figure 3.4. Referring to the Traditional Chinese Multilingual Language Understanding leaderboard TMMLU+ [27] , we tested models such as gpt-3.5, Taiwan-llama-v2.1-7B [21], Taiwan-llama-v2.0-13B [21], TAIDE-LX-7B-Chat [23], and MediaTek-Research/Breeze-7B-Instruct-v0.1[3]. To avoid sensitive political issues, we refrain from using models developed in China.

System Prompt: "你是一個人工智慧助理以及台灣大學校園導覽員"
Message Prompt: "以下是參考資料，請忽略不相關的文件，回答盡量簡短精要，切勿重複輸出一樣文句子：{retrieval documents}
請問：{paragraph}"

Figure 3.4: Prompt template for LLM generation.

doi:10.6342/NTU202400247

Chapter 4 Experiments

4.1 Automated Evaluation

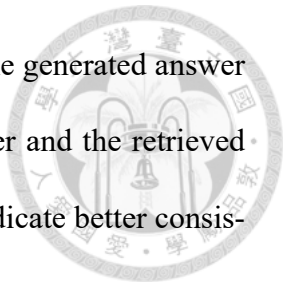
RAGAS [5] introduces three metrics to assess the performance of RAG systems. Below, we will explain the concepts of these three metrics. For implementation details, please refer to the RAGAS paper.

Context relevance. This metric assesses the relevance of the retrieved context, computed based on both the question and the contexts. The values range between 0 and 1, with higher values indicating superior relevancy.

Ideally, the retrieved context should solely encompass crucial information necessary to resolve the provided query. To calculate this, we first determine the value S by counting the number of sentences within the retrieved context that are pertinent for answering the given question, with the relevance of these sentences identified by the LLM. The final score is determined by the following formula, where F represents the total number of sentences in the retrieval context:

$$\text{Context relevance} = \frac{S}{F}$$

doi:10.6342/NTU202400247



Faithfulness. This metric evaluates the factual consistency of the generated answer with respect to the provided context. It is computed from the answer and the retrieved context, with the score scaled to the range of 0 to 1. Higher scores indicate better consistency.

The generated answer is considered faithful if all the assertions it makes can be inferred from the provided context. To calculate this, a set of claims from the generated answer is generated by the LLM. Each claim is then cross-checked by the LLM with the given context to ascertain if it can be inferred from the context. The final score is determined by the following formula, where C represents the number of claims in the answer that can be inferred from the given context, and A represents the total number of claims in the generated answer:

$$\text{Faithfulness} = \frac{C}{A}$$

Answer relevance. We assess the relevance of an answer based on how appropriately it addresses the question. Our assessment of answer relevance considers only reasonableness, not factuality, and penalizes cases where the answer is incomplete or contains redundant information.

To achieve this, we use the LLM to generate potential questions, denoted as q_i, that could have been derived from the answer (reverse-engineered). We then obtain embeddings for all these questions using BAAI/bge-large-zh. For each q_i, we calculate the similarity $\text{sim}(q, q_i)$ with the original question q as the cosine similarity between the corresponding embeddings. The answer relevance score for question q is then computed as:

$$\text{Answer relevance} = \frac{1}{n}\Sigma_{i=i}^{n} sim(q, q_i)$$

doi:10.6342/NTU202400247

In addition to the aforementioned metrics, an additional indicator was incorporated, as it is one of the most intuitive and classic ways to assess answer quality:

Answer similarity. Answer similarity refers to the assessment of the semantic resemblance between the generated answer and the ground truth. This evaluation is based on comparing the ground truth with the generated answer, with scores ranging from 0 to 1. A higher score indicates better alignment between the generated answer and the ground truth.

$$\text{Answer similarity} = \text{sim}(\text{ground truth}, \text{generated answer})$$

4.1.1 Evaluation of Answer Quality

The objective of this experimental section is to assess whether incorporating our method yields superior performance compared to solely utilizing LLMs. The methodology involves comparing the similarity between the ground truth and the answers generated by LLMs.

One approach evaluates Answer Relevance, considering the reasonableness of the response as proposed by RAGAS, while another approach, Answer Similarity, relies solely on the similarity between the ground truth answer and the generated answer.

From Table 4.1, it is evident that incorporating retrieval in RAG mostly outperforms the purely inference-based LLM model. Furthermore, we observe that models with initially stronger capabilities exhibit better overall performance than comparatively weaker models [27], thereby reaffirming the results of ICRALM [22].

doi:10.6342/NTU202400247

Model	Answer Relevancy	Answer Similarity
GPT-3.5 w/o RAG	**0.78**	0.69
GPT-3.5 w/ RAG	0.70	0.72
Taiwam-LLama v2.1 7B w/o RAG	0.58	0.63
Taiwam-LLama v2.1 7B w/ RAG	0.62	0.68
Taiwam-LLama v2.0 13B w/o RAG	0.69	0.68
Taiwam-LLama v2.0 13B w/ RAG	0.62	0.71
Breeze-7B-Instruct-v0.1 w/o RAG	0.73	0.68
Breeze-7B-Instruct-v0.1 w/ RAG	0.72	**0.73**
TAIDE-LX-7B-Chat w/o RAG	0.53	0.69
TAIDE-LX-7B-Chat w/ RAG	0.57	0.71

Table 4.1: Evaluation of answer quality.

4.1.2 Evaluation of Retrieval Quality and LLM Comprehension

This section investigates the quality of retrieved documents and assesses LLM comprehension. High faithfulness indicates that the generated answer aligns closely with the content of the document, ensuring accuracy and reliability. Context relevance measures how well the retrieved documents relate to the question, reflecting the overall quality and pertinence of the documents.

In Table 4.2, we observe that there is no significant difference in context relevance scores, which all remain low. This can be attributed to the considerable noise present on the web pages, which adversely affects context relevance. In contrast, faithfulness scores are high in general and we cab see there is notable 25 percent gap between the highest and lowest one. For instance, GPT-3.5's responses tend to retain more details from the document, while Taiwan-LLama's responses are often shorter. This disparity is likely due to differences in the models' inherent comprehension capabilities. Figure 4.1 illustrates these observations:

doi:10.6342/NTU202400247

Model	Faithfulness	Context Relevance
GPT-3.5 w/ RAG	0.68	0.354
Taiwan-LLama v2.1 7B w/ RAG	0.46	**0.359**
Taiwan-LLama v2.0 13B w/ RAG	0.62	0.357
Breeze-7B-Instruct-v0.1 w/ RAG	**0.71**	0.355
TAIDE-LX-7B-Chat w/ RAG	0.61	0.328

Table 4.2: Evaluation of retrieval quality and LLM comprehension.

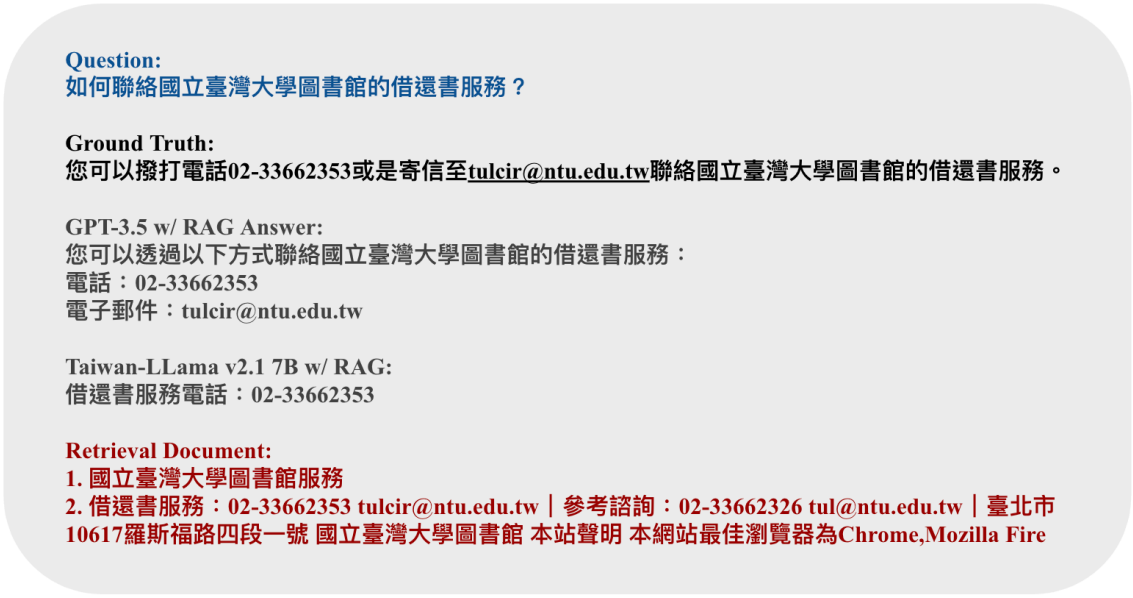


Figure 4.1: Comparison of response: GPT-3.5 vs. Taiwan-LLama v2.1 7B.

4.2 Human Evaluation

We designed a questionnaire to assess the performance of LLMs with RAG, focusing on three major indicators:

Answer similarity. Evaluates the similarity between the ground truth answer and the content of the language model's response.

Answer precision. Assesses the precision of the language model's response.

Model preferences. Measures subjective overall preferences.

The questionnaire consists of a total of 24 questions, distributed across three sections: 7 questions in the first section on Answer Similarity, 7 questions in the second section on

doi:10.6342/NTU202400247

Answer Precision, and 10 questions in the third section on Model Preferences. Ratings for the first two sections are provided on a scale of 1 to 5. The third section involves binary choice questions, where respondents select between two models.

In the questionnaire, there are four models, primarily divided into two main groups: **GPT-3.5 and GPT-3.5 with RAG.** This group aims to observe the potential benefits to users when RAG is integrated.

Taiwan-LLama with RAG (ours) and Taiwan-LLama with RAG (official website). This comparison evaluates whether our RAG method surpasses the current RAG integration on the official Taiwan-LLama website, which includes web search capabilities.

In total, we collected 35 survey responses. The final scores will undergo min-max normalization for each respondent in each part, allowing us to observe their distribution.

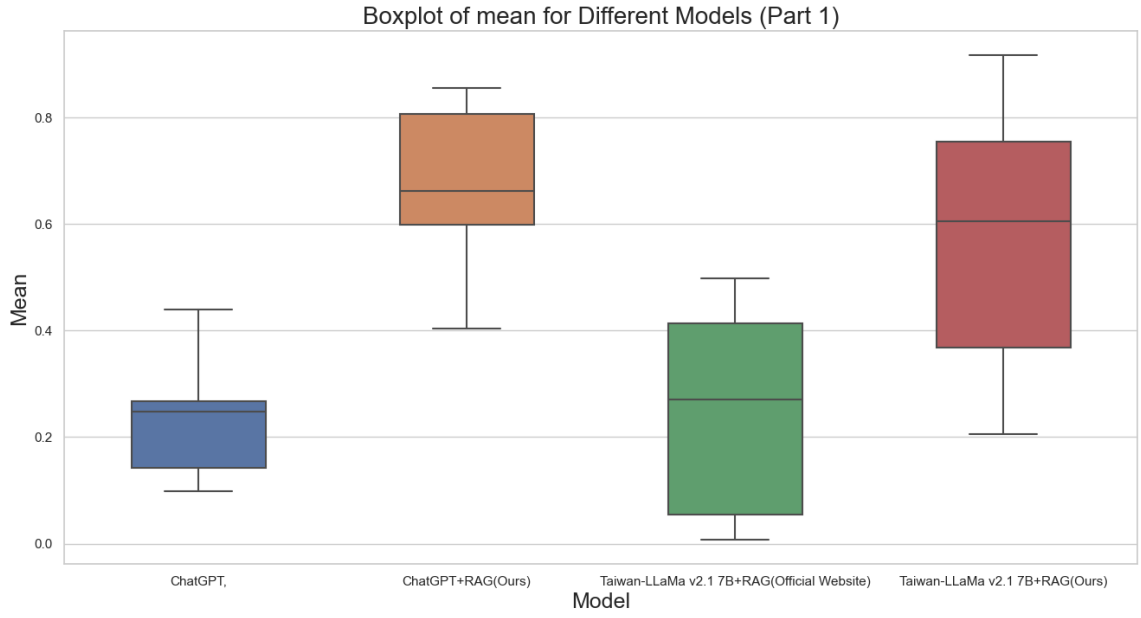


Figure 4.2: **Answer similarity.** Participants rate the similarity between the ground truth and the generated answer on a scale of 1 to 5 for 7 questions.To ensure consistency, the original scores will be normalized using min-max normalization, scaling them to a range of 0 to 1 for each respondent.

doi:10.6342/NTU202400247

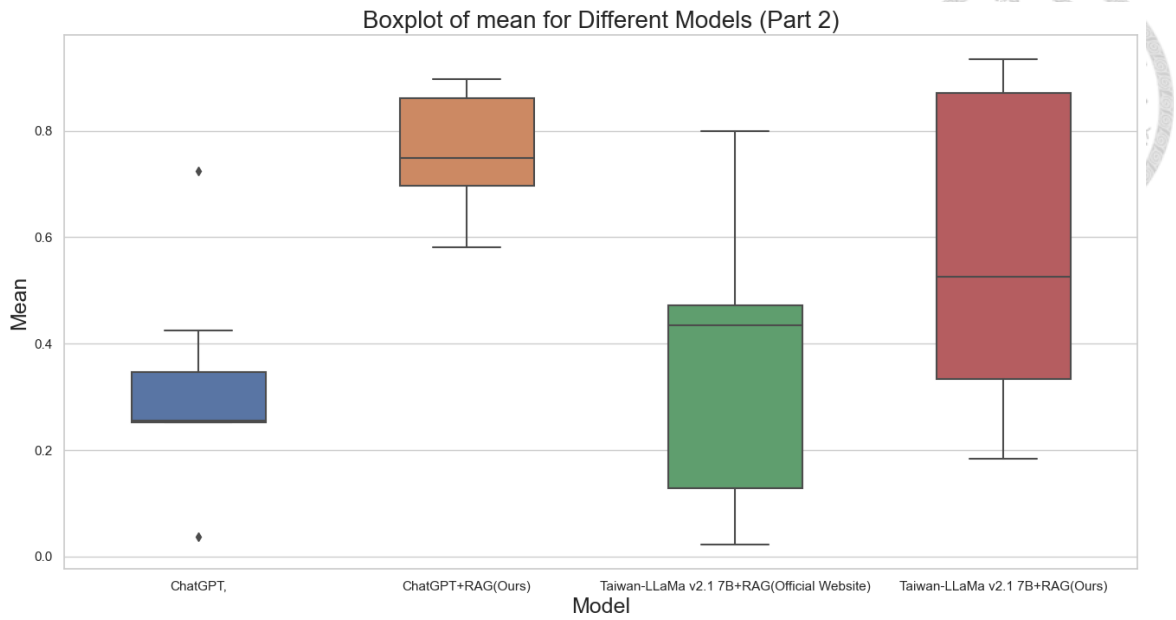


Figure 4.3: **Answer precision.** Participants rate the precision of the generated answers against the provided ground truth on a scale of 1 to 5 for 7 questions. To ensure consistency, the original scores will be normalized using min-max normalization, scaling them to a range of 0 to 1 for each respondent.

4.2.1 Answer Similarity

This section corresponds to the automated evaluation metric answer similarity. Participants are requested to directly assess the similarity between the ground truth and the generated answer, rating it on a scale from 1 to 5. From Figure 4.2, we observe that GPT-3.5 with RAG has the highest average score, followed by Taiwan-LLama with RAG (ours), Taiwan-LLama with RAG (official website), and pure GPT-3.5, in descending order. The scores indicate that any version incorporating RAG surpasses the solely inference-based LLM, demonstrating that RAG indeed enhances answer fidelity and user assistance.

4.2.2 Answer Precision

In the second part, users are asked to evaluate not only the similarity between the generated answer and the ground truth but also to assign high scores if the answer, though

doi:10.6342/NTU202400247

different from the ground truth, accurately addresses the question. Unlike the 7 questions in the previous section, the average ranking and distribution pattern observed in Figure 4.3. Answers provided by GPT-3.5 with RAG are noted as the most precise. Furthermore, users in the RAG group consistently rate their answers highly, demonstrating the stability of RAG across both the GPT-3.5 and Taiwan-LLama models.

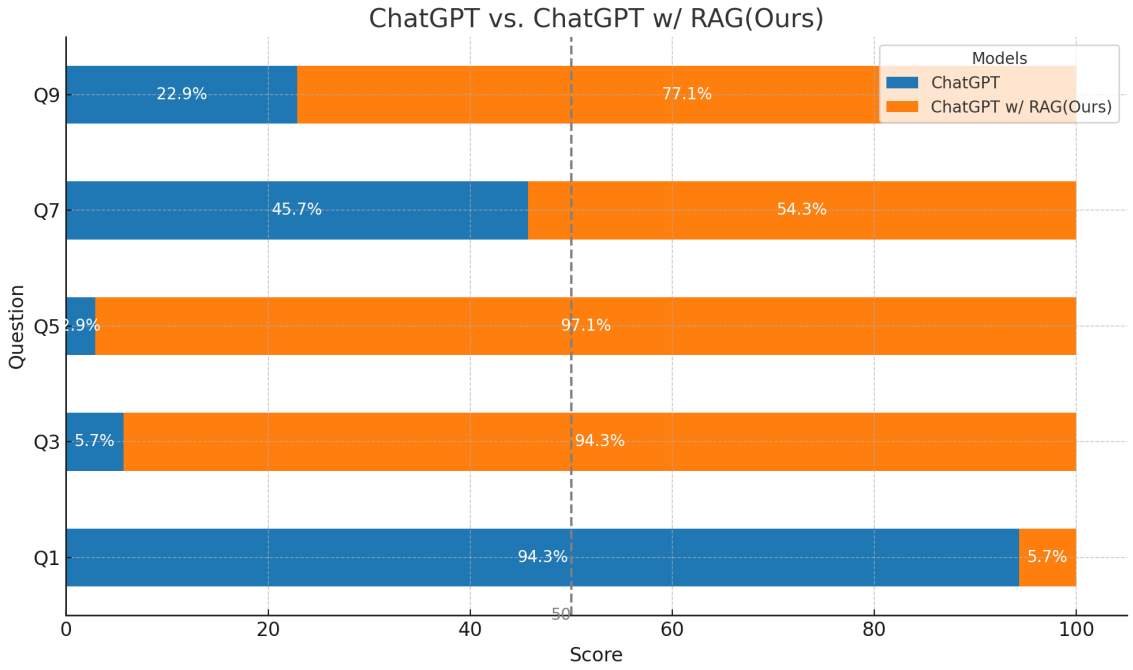


Figure 4.4: **GPT-3.5 group.** Each horizontal bar represents the percentage of individuals who prefer the output of that model in that question.

4.2.3 Model Preferences

In the third part, we divided the four models into two groups. Participants were asked to choose between liking the outputs of GPT-3.5 and GPT-3.5 with RAG, or preferring Taiwan-LLama with RAG (ours) or Taiwan-LLama with RAG (official website). Each group had five single-choice questions.

GPT-3.5 group. In Figure 4.4, ratio in this group is 4:1 in favor of GPT-3.5 with RAG over GPT-3.5 without RAG. This indicates that in the majority of cases, employing

doi:10.6342/NTU202400247

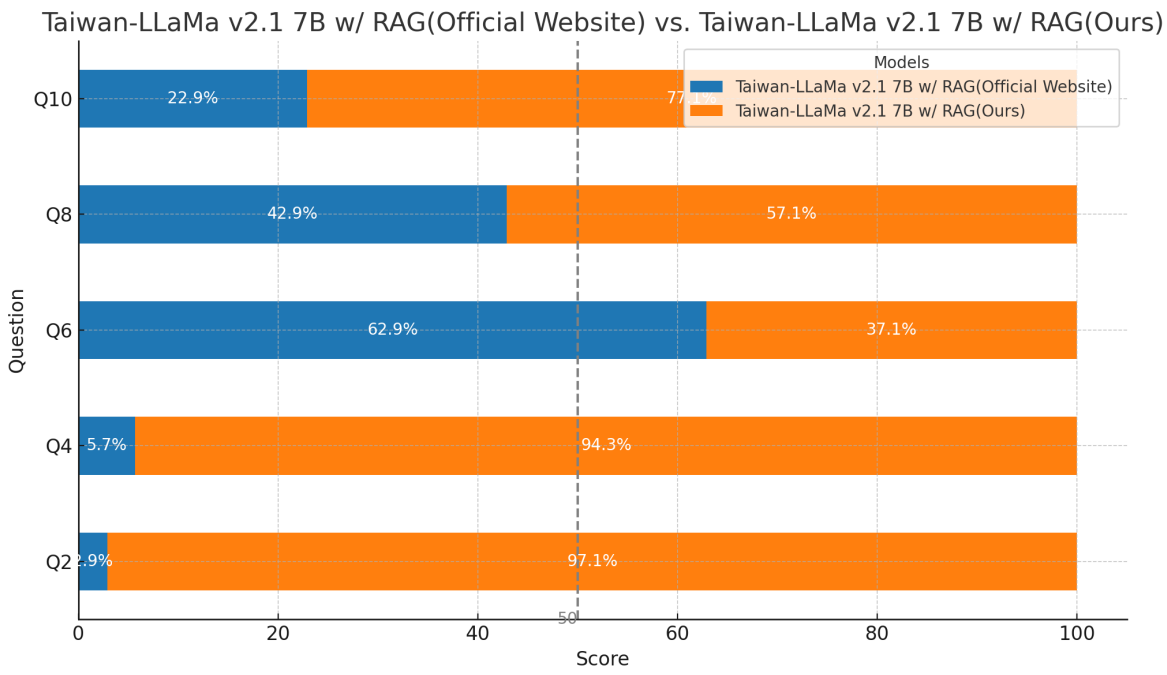


Figure 4.5: **Taiwan-LLama group.** Each horizontal bar represents the percentage of individuals who prefer the output of that model in that question.

RAG makes the answers more favored by users.

Taiwan-LLama group. In Figure 4.5, ratio in this group is 4:1 in favor of Taiwan-LLama with RAG (ours) over Taiwan-LLama with RAG (official website). This underscores that, in the majority of cases, our RAG method is more accurate and preferred by users.

4.3 Ablation Study

The term recall@K refers to the proportion of relevant documents retrieved among the top K documents for a given question. In our dataset, each question corresponds to only one document. Due to variations in document segmentation, the URL of the document retrieved by the question serves as the ground truth. In other words, if one of the URLs among the top K documents matches the ground truth URL, the recall is 1; otherwise, it

doi:10.6342/NTU202400247

CHUNK SIZE	Recall@5	Recall@10	Recall@20	Recall@50	Recall@100
GPT-3.5	**0.57**	**0.61**	**0.67**	**0.74**	**0.78**
32	0.56	0.61	0.65	0.70	0.76
64	0.55	0.60	0.65	0.72	0.76
128	0.55	0.59	0.64	0.70	0.76
256	0.54	0.60	0.64	0.71	0.75

Table 4.3: **Chunk optimization.** The objective of this experiment is to assess the optimal method for chunking documents. Recall@K signifies the proportion of relevant documents retrieved within the top K documents for a given question in our dataset, where each question corresponds to a single document.

Model	Recall@5	Recall@10	Recall@20	Recall@50	Recall@100
TownsWu/PEG	0.29	0.32	0.36	0.41	0.46
thenlper/gte-large-zh	0.49	0.52	0.57	0.64	0.69
BAAI/bge-large-zh-v1.5	**0.57**	**0.61**	**0.67**	**0.74**	**0.78**
infgrad/stella-large-zh	0.44	0.50	0.55	0.61	0.65
thenlper/gte-base-zh	0.53	0.57	0.62	0.69	0.73
BAAI/bge-base-zh-v1.5	0.54	0.60	0.66	0.73	0.77
infgrad/stella-base-zh	0.47	0.53	0.58	0.64	0.68
yentinglin/bert-base-zhtw	0.11	0.13	0.15	0.18	0.21
thenlper/gte-small-zh	0.53	0.59	0.64	0.71	0.75
BAAI/bge-small-zh-v1.5	0.51	0.55	0.61	0.67	0.71

Table 4.4: **Embedding model test.** The aim of this experiment is to evaluate which method is more suitable for vectorizing our chunks. Recall@K signifies the proportion of relevant documents retrieved within the top K documents for a given question in our dataset, where each question corresponds to a single document.

is 0. The following scores in table represent the average scores across all 996 questions in the test set.

Chunk optimization. Table 4.3 demonstrates the retrieval document recall experiment using the different chunk optimization strategy. In this experiment, GPT-3.5 segments the text into the smallest units containing complete semantic propositions and filters out webpage noise. This approach consistently retrieves the highest number of documents across all the K retrieval document settings.

Embedding model test. Table 4.4 presents the most suitable vectorization methods

doi:10.6342/NTU202400247

for the task at hand. Except for the BERT model mentioned exclusively in the table, which is trained using the Masked Language Model (MLM) approach, the remaining embedding models are trained using contrastive learning. The large model typically has a parameter count of around 330 million, the base model around 110 million, and the small model around 30 million. The inference time required is approximately in the ratio of 2.5:1.7:1 for large, base, and small models, respectively. Interestingly, the GBE series of models outperform the GTE series in terms of performance at the base and large levels. However, in the case of small models, the GTE series emerges victorious across the board. GTE-small is particularly suitable for applications that are time-sensitive.

4.4 Implementation Detail

Except for the inference GPT-3.5 using the OpenAI API, all experiments were conducted on a NVIDIA GeForce RTX 3090. Regardless of whether RAG was combined or not, the system prompt was set to: ”你是一個人工智慧助理以及台灣大學校園導覽員” to avoid unfairness caused by incomplete background descriptions in the questions.

doi:10.6342/NTU202400247

Chapter 5 Conclusion

In this study, we have developed a comprehensive system demonstration covering data acquisition, process design, testing, and final application for users. Our experiments have showcased that the combination of GPT3.5 with RAG, alongside the chunk optimization strategy proposed using GPT-3.5, and the BGE-Large Embedding model, yielded the most effective results. By harnessing RAG, we empower large language models to address challenges associated with long-tail and up-to-date information. Tailored for the domain of National Taiwan University, our application exemplifies a successful integration of large language models within a specific context.

Looking ahead, with the advancement of multi-modal capabilities, we anticipate exploring the replacement of the current purely text-based large language model. This advancement will enable our application to handle and generate a wider array of data types, thus enhancing its overall utility and versatility.

doi:10.6342/NTU202400247

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.

[3] F.-T. L. P.-C. H. Y.-C. C. D.-S. S. Chan-Jan Hsu, Chang-Le Liu. Breeze-7b-instruct-v0$_1$. $Accessed : 2024 - 01 - 25$.

[4] T. Chen, H. Wang, S. Chen, W. Yu, K. Ma, X. Zhao, D. Yu, and H. Zhang. Dense x retrieval: What retrieval granularity should we use? arXiv preprint arXiv:2312.06648, 2023.

[5] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert. Ragas: Automated evaluation of retrieval augmented generation. arXiv preprint arXiv:2309.15217, 2023.

[6] P. Finardi, L. Avila, R. Castaldoni, P. Gengo, C. Larcher, M. Piau, P. Costa, and V. Caridá. The chronicles of rag: The retriever, the chunk and the generator. arXiv preprint arXiv:2401.07883, 2024.

doi:10.6342/NTU202400247

[7] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, and H. Wang. Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997, 2023.

[8] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang. Retrieval augmented language model pre-training. In International conference on machine learning, pages 3929–3938. PMLR, 2020.

[9] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding, 2021.

[10] H. Huang, T. Tang, D. Zhang, W. X. Zhao, T. Song, Y. Xia, and F. Wei. Not all languages are created equal in llms: Improving multilingual capability by cross-lingual-thought prompting, 2023.

[11] infgrad. stella-base-zh. Accessed: 2024-01-25.

[12] infgrad. stella-large-zh. Sep 11, 2023.

[13] G. Izacard, M. Caron, L. Hosseini, S. Riedel, P. Bojanowski, A. Joulin, and E. Grave. Unsupervised dense information retrieval with contrastive learning. arXiv preprint arXiv:2112.09118, 2021.

[14] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mixtral of experts, 2024.

[15] N. Kandpal, H. Deng, A. Roberts, E. Wallace, and C. Raffel. Large language models struggle to learn long-tail knowledge, 2023.

doi:10.6342/NTU202400247

[16] V. Karpukhin, B. Oğuz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:2004.04906, 2020.

[17] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in Neural Information Processing Systems, 33:9459–9474, 2020.

[18] X. Li, E. Nie, and S. Liang. From classification to generation: Insights into crosslingual retrieval augmented icl. arXiv preprint arXiv:2311.06595, 2023.

[19] Z. Li, X. Zhang, Y. Zhang, D. Long, P. Xie, and M. Zhang. Towards general text embeddings with multi-stage contrastive learning. arXiv preprint arXiv:2308.03281, 2023.

[20] Z. Li, X. Zhang, Y. Zhang, D. Long, P. Xie, and M. Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023.

[21] Y.-T. Lin and Y.-N. Chen. Taiwan llm: Bridging the linguistic divide with a culturally aligned language model, 2023.

[22] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham. In-context retrieval-augmented language models. arXiv preprint arXiv:2302.00083, 2023.

[23] S. . T. P. Research and I. C. (NARLabs). taide/taide-lx-7b-chat. Accessed: 2024-07-30.

[24] S. Robertson and H. Zaragoza. The probabilistic relevance framework: Bm25 and beyond. Foundations and Trends® in Information Retrieval, 3(4):333–389, 2009.

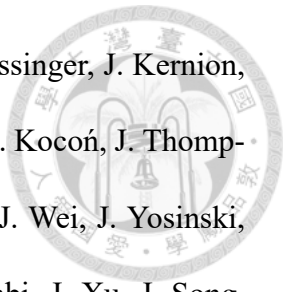
[25] J. Saad-Falcon, O. Khattab, C. Potts, and M. Zaharia. Ares: An automated evaluation

doi:10.6342/NTU202400247

framework for retrieval-augmented generation systems. arXiv preprint arXiv:2311.09476, 2023.

[26] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, A. Kluska, A. Lewkowycz, A. Agarwal, A. Power, A. Ray, A. Warstadt, A. W. Kocurek, A. Safaya, A. Tazarv, A. Xiang, A. Parrish, A. Nie, A. Hussain, A. Askell, A. Dsouza, A. Slone, A. Rahane, A. S. Iyer, A. Andreassen, A. Madotto, A. Santilli, A. Stuhlmüller, A. Dai, A. La, A. Lampinen, A. Zou, A. Jiang, A. Chen, A. Vuong, A. Gupta, A. Gottardi, A. Norelli, A. Venkatesh, A. Gholamidavoodi, A. Tabassum, A. Menezes, A. Kirubarajan, A. Mullokandov, A. Sabharwal, A. Herrick, A. Efrat, A. Erdem, A. Karakaş, B. R. Roberts, B. S. Loe, B. Zoph, B. Bojanowski, B. Özyurt, B. Hedayatnia, B. Neyshabur, B. Inden, B. Stein, B. Ekmekci, B. Y. Lin, B. Howald, B. Orinion, C. Diao, C. Dour, C. Stinson, C. Argueta, C. F. Ramírez, C. Singh, C. Rathkopf, C. Meng, C. Baral, C. Wu, C. Callison-Burch, C. Waites, C. Voigt, C. D. Manning, C. Potts, C. Ramirez, C. E. Rivera, C. Siro, C. Raffel, C. Ashcraft, C. Garbacea, D. Sileo, D. Garrette, D. Hendrycks, D. Kilman, D. Roth, D. Freeman, D. Khashabi, D. Levy, D. M. González, D. Perszyk, D. Hernandez, D. Chen, D. Ippolito, D. Gilboa, D. Dohan, D. Drakard, D. Jurgens, D. Datta, D. Ganguli, D. Emelin, D. Kleyko, D. Yuret, D. Chen, D. Tam, D. Hupkes, D. Misra, D. Buzan, D. C. Mollo, D. Yang, D.-H. Lee, D. Schrader, E. Shutova, E. D. Cubuk, E. Segal, E. Hagerman, E. Barnes, E. Donoway, E. Pavlick, E. Rodola, E. Lam, E. Chu, E. Tang, E. Erdem, E. Chang, E. A. Chi, E. Dyer, E. Jerzak, E. Kim, E. E. Manyasi, E. Zheltonozhskii, F. Xia, F. Siar, F. Martínez-Plumed, F. Happé, F. Chollet, F. Rong, G. Mishra, G. I. Winata, G. de Melo, G. Kruszewski, G. Parascandolo, G. Mariani, G. Wang, G. Jaimovitch-López, G. Betz, G. Gur-Ari, H. Galijasevic, H. Kim, H. Rashkin, H. Hajishirzi, H. Mehta, H. Bogar, H. Shevlin, H. Schütze,

doi:10.6342/NTU202400247



H. Yakura, H. Zhang, H. M. Wong, I. Ng, I. Noble, J. Jumelet, J. Geissinger, J. Kernion, J. Hilton, J. Lee, J. F. Fisac, J. B. Simon, J. Koppel, J. Zheng, J. Zou, J. Kocoń, J. Thompson, J. Wingfield, J. Kaplan, J. Radom, J. Sohl-Dickstein, J. Phang, J. Wei, J. Yosinski, J. Novikova, J. Bosscher, J. Marsh, J. Kim, J. Taal, J. Engel, J. Alabi, J. Xu, J. Song, J. Tang, J. Waweru, J. Burden, J. Miller, J. U. Balis, J. Batchelder, J. Berant, J. Frohberg, J. Rozen, J. Hernandez-Orallo, J. Boudeman, J. Guerr, J. Jones, J. B. Tenenbaum, J. S. Rule, J. Chua, K. Kanclerz, K. Livescu, K. Krauth, K. Gopalakrishnan, K. Ignatyeva, K. Markert, K. D. Dhole, K. Gimpel, K. Omondi, K. Mathewson, K. Chiafullo, K. Shkaruta, K. Shridhar, K. McDonell, K. Richardson, L. Reynolds, L. Gao, L. Zhang, L. Dugan, L. Qin, L. Contreras-Ochando, L.-P. Morency, L. Moschella, L. Lam, L. Noble, L. Schmidt, L. He, L. O. Colón, L. Metz, L. K. Şenel, M. Bosma, M. Sap, M. ter Hoeve, M. Farooqi, M. Faruqui, M. Mazeika, M. Baturan, M. Marelli, M. Maru, M. J. R. Quintana, M. Tolkiehn, M. Giulianelli, M. Lewis, M. Potthast, M. L. Leavitt, M. Hagen, M. Schubert, M. O. Baitemirova, M. Arnaud, M. McElrath, M. A. Yee, M. Cohen, M. Gu, M. Ivanitskiy, M. Starritt, M. Strube, M. Swędrowski, M. Bevilacqua, M. Yasunaga, M. Kale, M. Cain, M. Xu, M. Suzgun, M. Walker, M. Tiwari, M. Bansal, M. Aminnaseri, M. Geva, M. Gheini, M. V. T, N. Peng, N. A. Chi, N. Lee, N. G.-A. Krakover, N. Cameron, N. Roberts, N. Doiron, N. Martinez, N. Nangia, N. Deckers, N. Muennighoff, N. S. Keskar, N. S. Iyer, N. Constant, N. Fiedel, N. Wen, O. Zhang, O. Agha, O. Elbaghdadi, O. Levy, O. Evans, P. A. M. Casares, P. Doshi, P. Fung, P. P. Liang, P. Vicol, P. Alipoormolabashi, P. Liao, P. Liang, P. Chang, P. Eckersley, P. M. Htut, P. Hwang, P. Miłkowski, P. Patil, P. Pezeshkpour, P. Oli, Q. Mei, Q. Lyu, Q. Chen, R. Banjade, R. E. Rudolph, R. Gabriel, R. Habacker, R. Risco, R. Millière, R. Garg, R. Barnes, R. A. Saurous, R. Arakawa, R. Raymaekers, R. Frank, R. Sikand, R. Novak, R. Sitelew, R. Le-

doi:10.6342/NTU202400247

Bras, R. Liu, R. Jacobs, R. Zhang, R. Salakhutdinov, R. Chi, R. Lee, R. Stovall, R. Teehan, R. Yang, S. Singh, S. M. Mohammad, S. Anand, S. Dillavou, S. Shleifer, S. Wiseman, S. Gruetter, S. R. Bowman, S. S. Schoenholz, S. Han, S. Kwatra, S. A. Rous, S. Ghazarian, S. Ghosh, S. Casey, S. Bischoff, S. Gehrmann, S. Schuster, S. Sadeghi, S. Hamdan, S. Zhou, S. Srivastava, S. Shi, S. Singh, S. Asaadi, S. S. Gu, S. Pachchigar, S. Toshniwal, S. Upadhyay, Shyamolima, Debnath, S. Shakeri, S. Thormeyer, S. Melzi, S. Reddy, S. P. Makini, S.-H. Lee, S. Torene, S. Hatwar, S. Dehaene, S. Divic, S. Ermon, S. Biderman, S. Lin, S. Prasad, S. T. Piantadosi, S. M. Shieber, S. Misherghi, S. Kiritchenko, S. Mishra, T. Linzen, T. Schuster, T. Li, T. Yu, T. Ali, T. Hashimoto, T.-L. Wu, T. Desbordes, T. Rothschild, T. Phan, T. Wang, T. Nkinyili, T. Schick, T. Kornev, T. Tunduny, T. Gerstenberg, T. Chang, T. Neeraj, T. Khot, T. Shultz, U. Shaham, V. Misra, V. Demberg, V. Nyamai, V. Raunak, V. Ramasesh, V. U. Prabhu, V. Padmakumar, V. Srikumar, W. Fedus, W. Saunders, W. Zhang, W. Vossen, X. Ren, X. Tong, X. Zhao, X. Wu, X. Shen, Y. Yaghoobzadeh, Y. Lakretz, Y. Song, Y. Bahri, Y. Choi, Y. Yang, Y. Hao, Y. Chen, Y. Belinkov, Y. Hou, Y. Hou, Y. Bai, Z. Seid, Z. Zhao, Z. Wang, Z. J. Wang, Z. Wang, and Z. Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, 2023.

[27] Z.-R. Tam and Y.-T. Pai. An improved traditional chinese evaluation suite for foundation model. arXiv, 2023.

[28] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288, 2023.

[29] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal. Interleaving retrieval with

doi:10.6342/NTU202400247

chain-of-thought reasoning for knowledge-intensive multi-step questions. arXiv preprint arXiv:2212.10509, 2022.

[30] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. Advances in neural information processing systems, 32, 2019.

[31] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei. Improving text embeddings with large language models. arXiv preprint arXiv:2401.00368, 2023.

[32] T. Wu, Y. Qin, E. Zhang, Z. Xu, Y. Gao, K. Li, and X. Sun. Towards robust text retrieval with progressive learning, 2023.

[33] S. Xiao, Z. Liu, P. Zhang, and N. Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023.

[34] yentinglin. bert-base-zhtw. Accessed: 2024-01-25.

[35] W. Yu, D. Iter, S. Wang, Y. Xu, M. Ju, S. Sanyal, C. Zhu, M. Zeng, and M. Jiang. Generate rather than retrieve: Large language models are strong context generators, 2023.

[36] Y. Zhang, Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen, L. Wang, A. T. Luu, W. Bi, F. Shi, and S. Shi. Siren's song in the ai ocean: A survey on hallucination in large language models, 2023.

doi:10.6342/NTU202400247