SHARIF UNIVERSITY OF TECHNOLOGY

SCHOOL OF COMPUTER ENGINEERING

COURSE:

Natural language Processing

PROJECT TITLE:

Persian legal assistant system based on RAG integrated with LLM

TEAM MEMBERS:

Arash Tavangar, Amirtaha Amanzadi
Arash Alikhani, Shima Rezaei
Shayan Mohammadizadeh

COURSE PROFESSOR:

Ehsaneddin Asgari

ESFAND 1402

1 Abstract

In the complex field of law, there is a pressing need for effective legal information retrieval and comprehension, specially for languages other than English. This project attempts to meet that requirement by implementing such system for the official laws of Islamic Republic of Iran which are written in Persian. The system creates a pipeline that utilizes a combination of large language models (LLMs), text chunking, and document retrieval to deliver contextually relevant and nuanced responses to legal queries, acknowledging the difficulties involved in comprehending complicated legal texts. A key innovation lies in the Retrieval-Augmented Generation (RAG) configuration, which enriches the depth of generated answers by incorporating relevant articles of law for each query. In addition to having access to updated and extensive domain specific data, the integration of RAG prevents LLMs from hallucinating which is a common issue. RAG implementation with three different word embedding (OpenAI, LaBSE, fastText) were evaluated and compared with GPT-3.5 turbo acting as generator. The results yield OpenAI embedding model performs best in most metrics, while using LaBSE embedding model also achieved satisfactory results.

2 Introduction

The legal industry has historically been characterized by its exclusivity and the high cost of obtaining expert legal advice. Due to this exclusivity, small businesses and individuals frequently lack access to affordable legal representation, which could disadvantage them in court.

With the development of machine learning and artificial intelligence, there is a rare chance to bridge this gap by using an AI-powered assistant to deliver expert-level legal advice. It can potentially produce accurate, contextually appropriate legal advice across a range of legal disciplines and jurisdictions because of its extensive RAG querying capabilities. Furthermore, a lack of previous work on using LLMs alongside RAG within the legal domain in Persian language, was a great motivation for this project.

Despite their benefits, dealing with LLMs presents a number of difficulties, including factuality problems, hallucinations, and gaps in domain knowledge and outdated information. [1] Retrieval Augmented Generation offers a way to mitigate some of these problems by augmenting LLMs with external knowledge such as databases. When it comes to scenarios requiring a lot of expertise or when domain-specific apps need access to constantly updated information, RAG is very helpful. One important benefit of RAG over other methods is that task-specific applications do not require retraining of the LLM.

The integration of LLM with RAG can guarantee that answers are firmly based in the legal knowledge encoded in the vector database, in addition to being grammatically coherent. The initiative is significant because it has the potential to simplify legal research and increase accessibility for researchers, legal practitioners, and the general public. For individuals looking for clarification on Iranian Law, the system proves to be a useful tool as it tackles the difficulties associated with comprehending and navigating intricate legal paperwork.

2.1 Related work

As mentioned above, there are close to none previous works that tackle our specific problem in Persian; which is our main point of difference. However, there exist successful attempts in other languages predominantly English, most of which are commercial products, such as CoCouncil [2] and Lexis+ AI [3].

CoCounsels AI assistant is a recent mainstream product from Thomson Reuters. Amongst other practices, It utilizes RAG to prevent the large language models (LLMs) from making up things like case names and citations by focusing the LLMs on the actual language of Westlaw [1] content.

A non-English academical example is the work of Saud S. et al [4]. They proposed a question answering (QA) system, based on RAG transformer model on the Islamic law (Sharia law) in Arabic: Knowledge Augmented BERT2BERT (KAB). In contrast to our work which uses a general RAG approach that answers by directly searching for the relevant law articles to the question, KAB was trained of Frequently Asked Question (FAQ) dataset and returned the answer based on the nearest FAQ.

The differentiating factors of our project is that it is based on Iranian law with a completely Persian pipeline. Additionally, our implementation do not require any training or fine-tuning of LLMs, therefore it is computationally efficient and can easily be reproduced contrary to commercial systems that require extensive resources for training the language model.

3 Methodology

3.1 Dataset

The data used for creating the vectorized databases for RAG where fetched from official portal of Islamic Republic of Iran's laws and regulations that is available here [5]. There are a total of 16 different text documents each containing all articles for a specific law: Business, labour, civil, criminal procedure, dispute resolution councils, check issuance, family support, Islamic penal, social security, public accounting, law of customary affairs, law of tenders, valued added tax law, law of ownership of apartments, law of registrations of documents and properties, law on enforcement of civil judgements.

Moreover, an evaluation QA dataset for business law was manually created which contains 85 question and answer pairs divided into three levels of complexity, facilitating a comprehensive evaluation of our system's performance. :

- Easy: 29 simple questions regarding only one article of law, usually with a short answer.
- Medium: 28 questions posed from 1 or 2 law articles with longer answers
- Hard: 28 questions posed from 2 or 3 law articles with longer answers, or that require more reasoning to answer.

3.2 Pipeline architecture and models

The proposed pipeline design for our implementation follows Figure 2. The steps are:

1. Input: A user question is given to the system as input. We refer to this as user query.

[1] An online legal research service and proprietary database for lawyers and legal professionals available in over 60 countries.

2. Indexing: The related law document is first partitioned into chunks. Each chunk was chosen to be one law article since it was more optimal than naive document splitting. Then, we generate embeddings of the chunks, and index them into a vector store. At inference, the query is also embedded in a similar way. We used *LaBSE* [6] from Huggingface as our embedding model that supports Persian. We also experimented with *fastText* [7] as a comparison basis for our embedding model.

3. Retrieval: The relevant document chunks are obtained by comparing the query against the indexed vectors. It returns the k=5 most relevant chunks (law articles in our case) based on similarity score which is then used as context for the LLM input in the next step.

4. Generation: The original query is coupled with the relevant chunks to provide more context. The LLM receives the combined text with a predefined prompt after which it generates a response in natural language, which is then ready to be sent to the user as the system's final output. Two LLMs were tested for generation: MaralGPT-7B and GPT 3.5-turbo. The latter was chosen for the final implementation as it generates more accurate answers with much less response time compared to the former.

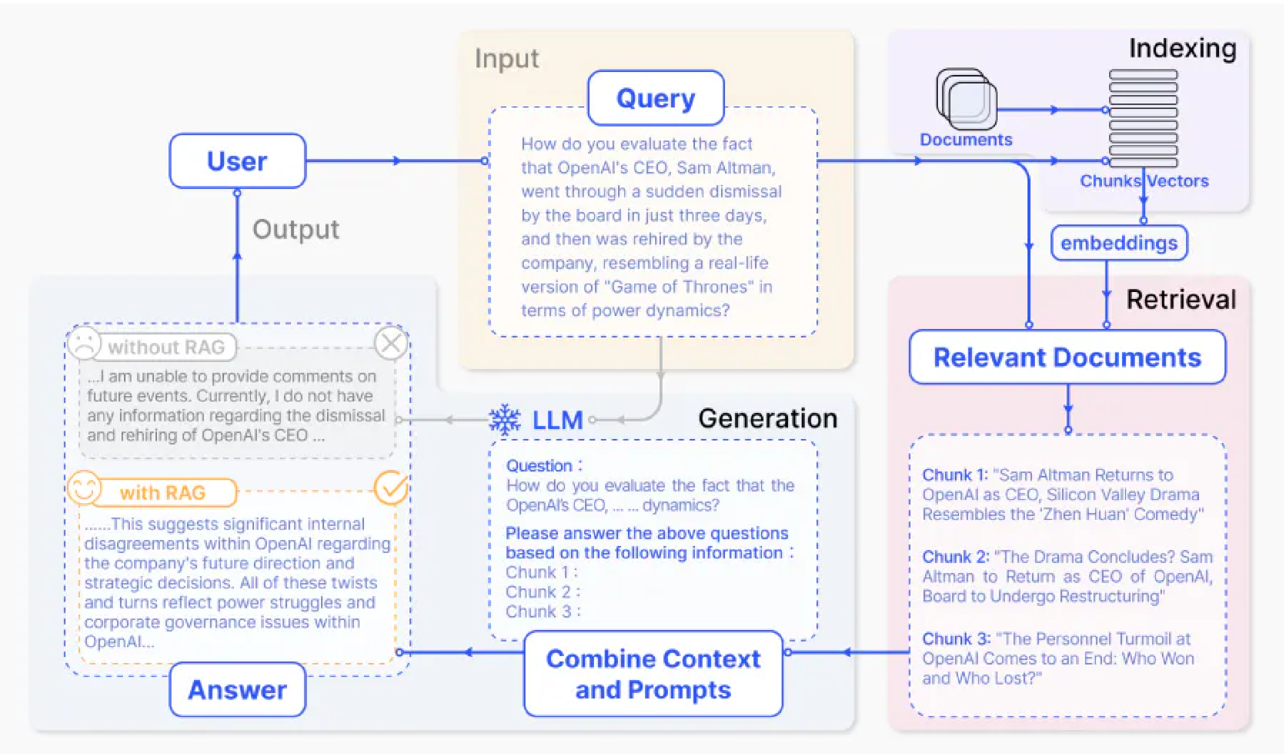

Figure 1: Example of RAG integrated with LLM pipeline [8]

Figure 2: Example of a QA for our implementation. The context is retrieved from relevant law articles to the query and given as extra information for the LLM prompt.

3.3 Experiments and evaluation method

The evaluation process for this project contained two components:

1. **Document retrieval evaluation:** The first evaluation component focuses on measuring the Mean Reciprocal Rank (MRR) score for document relevancy based on three distinct embedding functions. As each embedding function captures semantic information differently, assessing their efficacy in retrieving relevant documents is paramount. This evaluation aids in determining which embedding function yields deeper and more interpretable semantic information, thus influencing our system's overall performance.

2. **Final output evaluation using LLM:** The second evaluation component involves assessing the performance of the GPT-3.5 turbo in generating responses using our RAG system. Here, we compare the generated answers with our ground truth answers across different question types. By leveraging the document retrieval component and integrating it into the answer generation process, we evaluate the fidelity and relevance of the LLM-generated responses.

3.4 Dividing the work

There were 5 members for this team projects. The work was divided into three parts:

- Finding and creating the datasets: Two members were allocated to find as many laws and regulation documents from official sources as possible. Also, they manually created questions and answers related to business law that in three difficulties (easy, medium, and hard) which were used for evaluation purposes.

- Code implementation: Mostly two members were in charge of writing the code in Python, and deploying an user interface for presenting the project to other students.
- Report and presentation slides: One person was responsible for writing the project report and the presentation slides.

4 Evaluation results

4.1 Document Retrieval evaluation

In our evaluation phase, we primarily focus on measuring the Mean Reciprocal Rank (MRR) score in the document retrieval segment across different embedding models. MRR is a metric commonly used in information retrieval to assess the effectiveness of algorithms by considering the rank of the first relevant document retrieved for each query. The MRR score is calculated as the reciprocal of the rank of the first relevant document. We utilize MRR to assess the performance of our system under various settings, particularly with different values of K, where K denotes the number of documents retrieved for each query. Mathematically, it is defined as:

$$\text{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\text{rank}_i}$$

Where $|Q|$ is the total number of queries, and $rank_i$ represents the rank of the first relevant document retrieved for query q_i. Upon experimentation, we determined that $K = 5$ yields satisfactory results. We conduct evaluations on two distinct embedding models: fastText embedding and LaBSE embedding, and OpenAI embedding. According to table 1, LaBSE embedding model outperforms fastText, therefore it was chosen for the RAG during complete evaluation in section 4.2. OpenAI's embedding model full evaluation was difficult to conduct due to resource and time constraints. Given the nature of the questions posed in our dataset categorized as easy, medium, and hard the MRR scores are be higher for easier questions due to the straightforward nature of document relevancy. Further evaluation examples are provided in the appendix A.

Embedding model	Easy	Medium	Hard
fastText	0.203	0.095	0.107
LaBSE	**0.526**	**0.283**	**0.199**

Table 1: Average MMR of the embedding models used for document retrieval

4.2 Final output evaluation

In the second phase of evaluation, we focus on assessing the fidelity and relevance of the final output generated by the LLM based on the retrieved documents and the given question. We employ several metrics tailored to capture different aspects of the generated responses.

These metrics enable a comprehensive assessment of the quality and relevance of the generated responses, contributing to a deeper understanding of the system's performance. Elaborate description of these metrics is provided in appendix B:

1. **Faithfulness:** Insures the generated text stay true to the retrieved information.
2. **Answer Relevance:** Measures how pertinent the generated answer is to the given prompt.
3. **Context Recall:** Measures the extent to which the retrieved context aligns with the annotated answer, treated as the ground truth.
4. **Context Precision:** Evaluates whether all of the ground-truth relevant items present in the contexts are ranked higher or not.
5. **Answer Correctness:** Involves gauging the accuracy of the generated answer when compared to the ground truth.

Table 2 presents the final evaluation results obtained for each question type (easy, medium, and hard). The results include the performance metrics calculated for each category, providing insights into the system's efficacy across different levels of question complexity.

It is noteworthy that we opted for the OpenAI embedding function during the retrieval process due to its superior performance and enhanced embedding capabilities. However, considering the limitations of API usage, for simpler questions, we utilized LaBSE embedding, which outperformed fastText in terms of accuracy and efficiency. Additionally, it is essential to highlight that these evaluations are specifically conducted for business law questions. This focus was necessitated by our limited resources, which constrained the scope of our evaluation to this specific domain.

Question difficulty	Model (Embedding)	Faithfulness	Answer Relevance	Context Recall	Context Precision	Answer Correctness
Easy	GPT-3.5 (LaBSE)	0.887	0.735	0.84	0.675	0.606
	GPT-3.5 (LaBSE)+SC[2]	0.737	0.804	0.811	0.546	0.619
Medium	GPT-3.5 (OpenAI)	0.895	0.799	0.917	0.85	0.677
Hard	GPT-3.5 (OpenAI)	-	-	-	-	0.645

Table 2: Evaluation of final output (RAG+LLM) based on the selected metrics

The results diverge from our initial expectations. Interestingly, the performance for hard and medium questions surpasses that of easy questions. This variation could stem from several factors: firstly, the utilization of a less effective embedding function for easy questions might contribute to this discrepancy. Alternatively, easy questions, being more general in nature, might prompt the language model to generate answers that are less closely related to the query, thereby affecting the overall performance.

By meticulously evaluating the document retrieval and final output generation phases, we gain valuable insights into the strengths and limitations of our system, paving the way for future enhancements and optimizations in the domain of legal information retrieval and comprehension.

[2]Stands for Smart chunking which entails the segmentation of each document based on its individual rule number. Notably, only the first row for each question remains unaffected by smart chunking (referred to as random chunking), which tends to yield inferior results.

5 Conclusion and future work

Overall we can conclude that RAG infused with LLM that uses a suitable word embedding model that support Persian language can improve answering queries to users specifically in the law industry as they produce more reliable outputs. The chunking strategy can also be a important factor in performance.

There is great potential and opportunity for enhancing this project in future works. There already exist more advanced RAG implementations that take advantage of pre-retrieval and post-retrieval processing techniques. One such pre-retrieval technique that can be added is query rewriting that uses Query2Doc [9] method in order to align user query to documents in the semantic space. This will prove effective when a user's query may lack semantic information or contain imprecise phrasing. Information compression [10] and re-ranking [11] are also post-retrieval techniques that seem promising for future works. Furthermore, fine-tuning the LLM on Iranian legal law and fine-tuning the embedding model would most likely boost the performance, though they require more computing resources.

References

[1] P. Lewis, E. Perez, A. Piktus, *et al.*, *Retrieval-augmented generation for knowledge-intensive nlp tasks*, 2021. arXiv: `2005.11401 [cs.CL]`.

[2] *CoCouncil AI legal assistant*, 2024. [Online]. Available: `https://casetext.com/cocounsel`.

[3] *Lexis+ AI*, 2024. [Online]. Available: `https://www.lexisnexis.com/en-us/products/lexis-plus-ai.page`.

[4] S. S. Alotaibi, A. A. Munshi, A. T. Farag, O. E. Rakha, A. A. Al Sallab, and M. Alotaibi, "Kab: Knowledge augmented bert2bert automated questions-answering system for jurisprudential legal opinions," *International journal of computer science and network security: IJCSNS*, vol. 22, no. 6, pp. 346–356, 2022.

[5] [Online]. Available: `https://qavanin.ir`.

[6] *Sentence-transformers (LaBSE)*. [Online]. Available: `https://huggingface.co/sentence-transformers/LaBSE`.

[7] *Fasttext - Library for efficient representation learning*. [Online]. Available: `https://fasttext.cc`.

[8] *Retrieval Augmented Generation (RAG) for LLMs*, 2024. [Online]. Available: `https://www.promptingguide.ai/research/rag`.

[9] L. Wang, N. Yang, and F. Wei, *Query2doc: Query expansion with large language models*, 2023. arXiv: `2303.07678 [cs.IR]`.

[10] F. Xu, W. Shi, and E. Choi, *Recomp: Improving retrieval-augmented lms with compression and selective augmentation*, 2023. arXiv: `2310.04408 [cs.CL]`.

[11] *Advanced RAG 04: Re-ranking*. [Online]. Available: `https://pub.towardsai.net/advanced-rag-04-re-ranking-85f6ae8170b1`.

Appendices

A Examples of embedding models evaluation

Below are the MRR scores obtained for fastText model across different question types (with some examples for each of them and the average accuracy over all of them):

Easy questions :

Question	Answer	Related docs	Predicted docs	Accuracy(MRR)
شرایط تاجر بودن چیست	تاجر کسی است که شغل معمولی خود را معاملات تجارتی قرار دهد. (ماده ۱),	[1]	[450, 444, 101, 60, 517]	0
نحوه ثبت مراسلات و مخابرات در دفتر کپیه چگونه است	کلیه مراسلات و مخابرات و صورت‌حساب‌های صادره باید ترتیب تاریخ ثبت شود. (ماده ۱۰),	[10]	[10]	1

Medium questions :

Question	Answer	Related docs	Predicted docs	Accuracy(MRR)
تفاوت بین معاملات تجارتی که توسط تاجر انجام می‌شود و معاملاتی که برای رفع حوائج شخصی انجام می‌شوند چیست	معاملات تجارتی شامل خرید، فروش، دلالی و سایر فعالیت‌های تجاری است که به منظور کسب سود انجام می‌شوند، در حالی که معاملات برای رفع حوائج شخصی شامل فعالیت‌هایی نیست که به منظور کسب سود تجاری انجام می‌شوند. (ماده ۲ و ماده ۳),	[2, 3]	[335, 131, 133, 146, 367]	0
در شرکت مختلط غیر سهامی مسئولیت شریک ضامن و شریک با مسئولیت محدود چیست	شریک ضامن مسئول کلیه قروضی است که ممکن است علاوه بر دارائی شرکت پیدا شود - شریک با مسئولیت محدود کسی است که مسئولیت اوفقط تا میزان سرمایه است که در شرکت گذارده و یا بایستی بگذارد. (ماده ۱۴۱)	[141]	[150, 141]	0.5

Hard questions :

Question	Answer	Related docs	Predicted docs	Accuracy(MRR)
چه شرایطی برای تشکیل و ثبت یک شرکت با مسئولیت محدود وجود دارد و چگونه می‌توان اساسنامه آن را تغییر داد	برای تشکیل یک شرکت با مسئولیت محدود نیاز به پرداخت کامل سرمایه نقدی و تسلیم سهم‌الشرکه‌های غیر نقدی است. هرگونه تغییر در اساسنامه نیاز به اکثریت عددی شرکاء دارد. (مواد ۹۶، ۹۷ و ۱۱۱),	[96, 97, 111]	[241, 9, 170, 169, 141]	0
در چه صورتی تقسیم دارائی شرکتهای سهامی و شرکتهای با مسئولیت محدود و تعاونی بین شرکاء خواه در ضمن مدت تصفیه و خواه پس از ختم آن ممکن است؟	در صورتی که قبلاً سه مرتبه در مجله رسمی و یکی از جراید اعلان و یکسال از تاریخ انتشار اولین اعلان در مجله گذشته باشد. (مواد ۲۱۴ و ۲۱۵)	[214, 215]	[213, 214]	0.5

Now these are the MRR scores obtained for LaBSE model across different question types (with some examples for each of them and the average accuracy over all of them):

Easy questions :

Question	Answer	Related docs	Predicted docs	Accuracy(MRR)
شرایط تاجر بودن چیست	تاجر کسی است که شغل معمولی خود را معاملات تجارتی قرار دهد. (ماده ۱),	[1]	[1]	1
نحوه ثبت مراسلات و مخابرات در دفتر کپیه چگونه است	کلیه مراسلات و مخابرات و صورت‌حساب‌های صادره باید ترتیب تاریخ ثبت شود. (ماده ۱۰),	[10]	[10]	1

Medium questions :

Question	Answer	Related docs	Predicted docs	Accuracy(MRR)
تفاوت بین معاملات تجارتی که توسط تاجر انجام می‌شود و معاملاتی که برای رفع حوائج شخصی انجام می‌شوند چیست	معاملات تجارتی شامل خرید، فروش، دلالی و سایر فعالیت‌های تجاری است که به منظور کسب سود انجام می‌شوند، در حالی که معاملات برای رفع حوائج شخصی شامل فعالیت‌هایی نیست که به منظور کسب سود تجاری انجام می‌شوند. (ماده ۲ و ماده ۳),	[2, 3]	[1,3]	0.5
در شرکت مختلط غیر سهامی مسئولیت شریک ضامن و شریک با مسئولیت محدود چیست	شریک ضامن مسئول کلیه قروضی است که ممکن است علاوه بر دارائی شرکت پیدا شود - شریک با مسئولیت محدود کسی است که مسئولیت اوفقط تا میزان سرمایه است که در شرکت گذارده و یا بایستی بگذارد. (ماده ۱۴۱)	[141]	[164, 144, 156, 206, 162]	0

Hard questions :

Question	Answer	Related docs	Predicted docs	Accuracy(MRR)
چه شرایطی برای تشکیل و ثبت یک شرکت با مسئولیت محدود وجود دارد و چگونه می‌توان اساسنامه آن را تغییر داد	برای تشکیل یک شرکت با مسئولیت محدود نیاز به پرداخت کامل سرمایه نقدی و تسلیم سهم‌الشرکه‌های غیر نقدی است. هرگونه تغییر در اساسنامه نیاز به اکثریت عددی شرکاء دارد. (مواد ۹۶، ۹۷, و ۱۱۱)	[96, 97, 111]	[60, 174, 8, 69, 278]	0
در چه صورتی تقسیم دارائی شرکتهای سهامی و شرکتهای با مسئولیت محدود و تعاونی بین شرکاء خواه در ضمن مدت تصفیه و خواه پس از ختم آن ممکن است؟	در صورتی که قبلاً سه مرتبه در مجله رسمی و یکی از جراید اعلان و یکسال از تاریخ انتشار اولین اعلان در مجله گذشته باشد. (مواد ۲۱۴ و ۲۱۵)	[214, 215]	[215]	1

B Final output evaluation metrics

Faithfulness evaluates whether the LLM/generator in your RAG pipeline is generating LLM outputs that factually aligns with the information presented in the retrieval context.

$$\text{Faithfulness score} = \frac{|\text{Number of claims in the generated answer that can be inferred from given context}|}{|\text{Total number of claims in the generated answer}|}$$

Hint

Question: Where and when was Einstein born?

Context: Albert Einstein (born 14 March 1879) was a German-born theoretical physicist, widely held to be one of the greatest and most influential scientists of all time

High faithfulness answer: Einstein was born in Germany on 14th March 1879.

Low faithfulness answer: Einstein was born in Germany on 20th March 1879.

Answer Relevancy focuses on assessing how pertinent the generated answer is to the given prompt. A lower score is assigned to answers that are incomplete or contain redundant information. This metric is computed using the question and the answer, with values ranging between 0 and 1, where higher scores indicate better relevancy. An answer is deemed relevant when it directly and appropriately addresses the original question. Importantly, our assessment of answer relevance does not consider factuality but instead penalizes cases where the answer lacks completeness or contains redundant details.

To calculate this score, the LLM is prompted to generate an appropriate question for the generated answer multiple times, and the mean cosine similarity between these generated questions and the original question is measured. The underlying idea is that if the generated answer accurately addresses the initial question, the LLM should be able to generate questions from the answer that align with the original question.

Question: Where is France and what is its capital?
\- *Low relevance answer: France is in western Europe.*
\- *High relevance answer: France is in western Europe and Paris is its capital.*

Context recall measures the extent to which the retrieved context aligns with the annotated answer, treated as the ground truth. It is computed based on the ground truth and the retrieved context, and the values range between 0 and 1, with higher values indicating better performance.

To estimate context recall from the ground truth answer, each sentence in the ground truth answer is analyzed to determine whether it can be attributed to the retrieved context or not. In an ideal scenario, all sentences in the ground truth answer should be attributable to the retrieved context.

$$\text{context recall} = \frac{|\text{GT sentences that can be attributed to context}|}{|\text{Number of sentences in GT}|}$$

Context Precision is a metric that evaluates whether all of the ground-truth relevant items present in the contexts are ranked higher or not. Ideally all the relevant chunks must appear at the top ranks. This metric is computed using the question and the contexts, with values ranging between 0 and 1, where higher scores indicate better precision.

$$\text{Context Precision@k} = \frac{\sum \text{precision@k}}{\text{total number of relevant items in the top K results}}$$

$$\text{Precision@k} = \frac{\text{true positives@k}}{(\text{true positives@k} + \text{false positives@k})}$$

Where k is the total number of chunks in `contexts`

Question: Where is France and what is its capital?
Ground truth: France is in Western Europe and its capital is Paris.

- High context precision: [France, in Western Europe, encompasses medieval cities, alpine villages and Mediterranean beaches. Paris, its capital, is famed for its fashion houses, classical art museums including the Louvre and monuments like the Eiffel Tower, The country is also renowned for its wines and sophisticated cuisine. Lascauxs ancient cave drawings, Lyons Roman theater and the vast Palace of Versailles attest to its rich history.]

- Low context precision: [The country is also renowned for its wines and sophisticated cuisine. Lascauxs ancient cave drawings, Lyons Roman theater and, France, in Western Europe, encompasses medieval cities, alpine villages and Mediterranean beaches. Paris, its capital, is famed for its fashion houses, classical art museums including the Louvre and monuments like the Eiffel Tower,]

Answer Correctness involves gauging the accuracy of the generated answer when compared to the ground truth. This evaluation relies on the ground truth and the answer, with scores ranging from 0 to 1. A higher score indicates a closer alignment between the generated answer and the ground truth, signifying better correctness. Answer correctness encompasses two critical aspects: semantic similarity between the generated answer and the ground truth, as well as factual similarity. These aspects are combined using a weighted scheme to formulate the answer correctness score. Users also have the option to employ a threshold value to round the resulting score to binary, if desired.

Ground truth: Einstein was born in 1879 in Germany.
High answer correctness: In 1879, Einstein was born in Germany.
Low answer correctness: Einstein was born in Spain in 1879.

C Examples of final output evaluation

Here are some single examples on the evaluation dataset using GPT-3.5 tubro using LaBSE and OpenAI embedding model for RAG.

Easy Questions with LaBSE

	question	answer	contexts	ground_truth	faithfulness	answer_relevancy	context_recall	context_precision	answer_correctness
0	شرایط تاجر بودن چیست؟	شرایط تاجر بودن شامل انجام م	['ماده 1 - تاجر کسی است که 'ماده 3 - معاملات ذیل باعتبار	تاجر کسی است که شغل معمول	0	0.9662487552	1	1	0.8955666603
1	خرید یا تحصیل مال منقول برای	خرید یا تحصیل مال منقول برای	['ماده 4 - معاملات غیر منقول 'ماده 54(الحاقی 47,12,24 'ماده 2 - معاملات تجارتی از	برای فروش یا اجاره، چه تصر	1	0.9180358905	1	0.3333333333	0.7286394886
2	انواع عملیاتی که جزء معاملات	همه معاملات تجارتی محسوب	['ماده 4 - معاملات غیر منقول 'ماده 1 - تاجر کسی است که 'ماده 5 - کلیه معاملات تجار	عملیات دلالی، حق‌العمل‌کاری،	0.5	0.8924788376	1	0.3333333333	0.8575408286

Medium Questions with Open AI:

	question	answer	contexts	ground_truth	faithfulness	answer_relevanc	context_recall	context_precisio	answer_correctness
0	تفاوت بین معاملات تجا	تفاوت بین معاملات تجا	['ماده 3 - معاملات ذیل 'ماده 541 - تاجر در 'ماده 2 - معاملات تجا 'ماده 475 - حکم فوق 'ماده 474 - اگر اشخا	معاملات تجارتی شامل		0.9713325865	1	1	0.6099248325
1	چگونه دفتر روزنامه و	دفتر کل باید کلیه معاملا	['ماده 9 - دفتر دارائی 'ماده 8 - دفتر کل دفتر 'ماده 10 - دفتر کپیه د 'ماده 297 - دفتر محک 'ماده 7 - دفتر روزنامه	دفتر روزنامه باید روز		0.7901012061	1	1	0.7414486945
2	نقش دفتر دارایی و دفتر	دفتر دارایی برای ثبت	['ماده 9 - دفتر دارائی 'ماده 7 - دفتر روزنامه 'ماده 10 - دفتر کپیه د 'ماده 309 - تمام مقرر 'ماده 503 - هر گاه تا	دفتر دارایی صورت جا	1	0.8923898327	1	0.7555555555	0.7358999665