Real world RAG System

Objective:

Develop an efficient Retrieval-Augmented Generation (RAG) pipeline capable of delivering accurate answers to the user queries.

Dataset : https://huggingface.co/datasets/rungalileo/ragbench

Dataset Description : RAG Bench is a large-scale benchmark dataset specifically designed for training and evaluating RAG systems. It contains 100k examples spanning five industry-specific domains:

- Biomedical Research
- General Knowledge
- Legal
- Customer Support
- Finance

The dataset supports various RAG task types and includes evaluation metrics such as context relevance, context utilization, answer faithfulness and answer completeness.

The dataset samples are sourced from real-world industry corpora, such as user manuals, making the dataset highly relevant for practical applications. This comprehensive dataset enables a detailed assessment of RAG system performance, ensuring robust and reliable evaluation.

Project Overview: Generative large language models often face challenges such as producing outdated information or fabricating facts. Retrieval-Augmented Generation (RAG) techniques address these limitations by integrating pretraining and retrieval-based approaches. This combination creates a robust framework for improving model accuracy and reliability.

RAG offers the added benefit of enabling rapid deployment of domain-specific applications without requiring updates to the model parameters. As long as relevant documents are available for a query, the system can adapt to organizational or domain-specific needs efficiently.

A typical RAG workflow involves Query Classification, Retrieval, Reranking, Repacking and Summarization. Implementing RAG requires careful consideration of various factors, such as properly splitting documents into manageable chunks, selecting suitable embeddings to semantically represent these chunks, choosing vector databases for efficient storage and retrieval of feature representations.

By addressing these elements, RAG systems can deliver accurate, domain-specific, and context-aware responses, making them a powerful tool for real-world applications.

References:

1) RAGBench: Explainable Benchmark for Retrieval-Augmented Generation Systems

2) Searching for Best Practices in Retrieval-Augmented Generation,

Tools: Transformers, Hugging Face, PyTorch, Langchain, LlamaIndex

Deployments: FastAPI, Cloud Application Platform | Heroku, Streamlit, Cloud Computing, Hosting Services, and APIs | Google Cloud

Final Submissions:

- Project technical report & presentation with desired outcomes
- An overview of the modeling techniques used for the problem
- GitHub Repository of the project
- Summary of 3 research papers

Paper-2:

Benchmarking Large Language Models in Retrieval-Augmented Generation

Task

Analyze the performance of LLMs in 4 fundamental abilities required for RAG

1. Noise Robustness
2. Negative Rejection
3. Information Integration
4. Counterfactual Robustness

Abilities of LLM for RAG

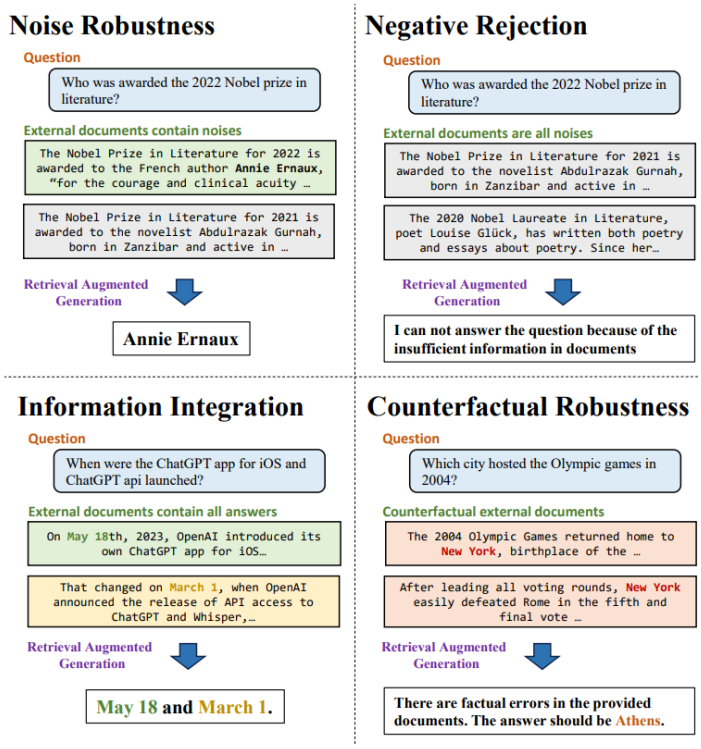

Figure 1: Illustration of 4 kinds of abilities required for retrieval-augmented generation of LLMs.

1. **Noise Robustness:** Ability of an LLM to extract the useful information in the presence of noisy documents
2. **Negative Rejection:** Ability of an LLM to reject answering the question when required knowledge is not present in any of the retrieved documents
3. **Information Integration:** Ability of an LLM to integrate information present in multiple documents to answer complex questions
4. **Counterfactual Robustness:** Evaluates whether LLMs can identify risks of known factual errors in the retrieved documents and generates correct answer using its pretrained knowledge

Dataset

RGB contains 600 base questions, and 200 additional questions for the information integration ability and 200 additional questions for counterfactual robustness ability. Half of the instances are in English, and the other half are in Chinese.

Language	*Noise Robustness Negative Rejection*	*Information Integration*	*Counterfactual Robustness*
English	300	100	100
Chinese	300	100	100

For Negative Rejection, all external documents are sampled from negative documents in Noise Robustness testbed.

Evaluation Metrics

- **Accuracy:** {*Noise Robustness, Information Integration*}
    - Uses exact matching approach: If the generated text contains an exact match of the answer, it is considered as a correct answer

- **Rejection Rate:** {*Negative Rejection*}
    - LLM should output the specific content “I can not answer the question because of insufficient information in documents” (we use instruction to inform the model)
    - If the model generates this content, it indicates a successful rejection

- **Error Detection Rate:** {*Counterfactual Robustness*}
    - LLM should output the specific content “There are factual errors in the provided documents” (we use instruction to inform the model)

    - If the model generates this content, it indicates the model has detected erroneous information in document

- **Error Correction Rate:** If the model generates correct answer then it is capable of correcting errors in documents

References

- Research Paper: Benchmarking Large Language Models in Retrieval-Augmented Generation

- Code and Data: https://github.com/chen700564/RGB