INSTITUIÇÃO EXECUTORA COORDENADORA APOIO

ACORDO DE PARCERIA Nº 05/23 FADE/UFPE/SOFTEX - RESIDENCIAL IC13 (CONVÊNIO Nº 02/2023 UFPE) 23076.125530/2022-28

Lux.AI

Avaliação de LLMs e Retrieval Para Aplicação de RAG no Mobile

---

INTRODUÇÃO

Nos últimos anos, o aumento significativo das pesquisas sobre Grandes Modelos de Linguagem (*Large Language Models* – LLM) levou ao surgimento de *chatbots* como ChatGPT, Gemini etc. Essas ferramentas, treinadas com um vasto conjunto de informações, conseguem auxiliar os usuários em diversas tarefas cotidianas, oferecendo respostas que rivalizam com as de humanos em diversas áreas do conhecimento. No entanto, os LLMs menores possuem limitações, como a incapacidade de acessar informações adquiridas após o treinamento e dificuldades em responder a questões pouco abordadas durante o processo de treinamento (Es et al., 2023). Para contornar esse problema, a utilização da Geração Aumentada de Recuperação (*Retrieval-Augmented Generation* - RAG) se apresenta como uma solução essencial. Esse método envolve a recuperação de informações relevantes para responder à questão em análise, buscando contextos mais adequados no conjunto de dados que expliquem de maneira satisfatória o que é solicitado (Lee et al., 2019; Guu et al., 2020).

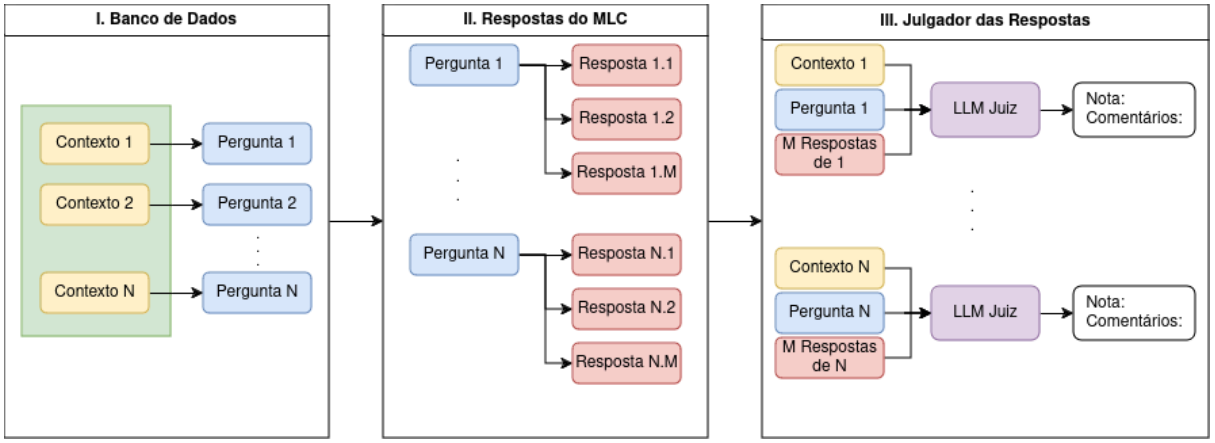
Neste contexto, o presente trabalho tem como objetivo desenvolver uma *pipeline* para execução em dispositivos móveis, visando extrair informações dos contextos disponibilizados por uma base de dados, além disso, será avaliada a eficácia da recuperação pelo RAG e também a qualidade da resposta gerada pelo LLM Gerador na *pipeline* do RAG. As respostas serão avaliadas por um LLM Juiz, que verificará se as respostas obtidas são satisfatórias, se correspondem ao texto base e a que classe de qualidade pertencem (Zheng et al., 2023; Saad-Falcon, 2023). Dessa forma, a aplicação deste trabalho permitirá ao usuário: i) criar uma base de dados de perguntas e respostas

com diversos contextos e/ou informações de diferentes campos do conhecimento; ii) avaliar a qualidade das respostas geradas pelos LLMs a partir de um LLM Juiz; iii) medir objetivamente a eficácia da recuperação pela RAG e a adequação da resposta ao contexto fornecido; iv) comparar diferentes LLMs Geradores para identificar aquele que melhor atende à recuperação do contexto descrito no documento de referência.

METODOLOGIA

Para avaliar o uso de LLMs aplicado com a recuperação de contexto, este tópico descreve a metodologia empregada para alcançar os resultados desejados. A Figura 1 ilustra todo o processo proposto para a aplicação desta metodologia. Resumidamente, o trabalho é dividido em três etapas: a construção de um banco de dados que relaciona o contexto disponível com uma pergunta; a obtenção de diferentes respostas geradas por um LLM Gerador para cada pergunta; e a avaliação de cada resposta utilizando um LLM Juiz. Os próximos tópicos detalham os passos envolvidos em cada uma dessas etapas.

Figura 1: Resumo da *pipeline* para a metodologia.



Fonte: O Autor (2024)

Banco de Dados

Imagine que um usuário tenha um conjunto de textos sobre um determinado tema, por exemplo, um manual de usuário dividido em tópicos, informações de diferentes áreas de conhecimento, como materiais de disciplinas de um curso. Como mencionado anteriormente, o uso de LLMs junto à pipelines de recuperação de dados (como o RAG) para auxiliar na extração de informações desses dados oferece maior facilidade ao buscar respostas para pontos específicos,

além de otimizar o tempo do usuário. No entanto, obter respostas precisas desses dados pode ser desafiador, especialmente se as informações específicas foram escassamente abordadas durante o treinamento do LLM (Azaria & Mitchell, 2023).

Para construção da base de dados é possível dividir o texto disponível em tópicos, ou "chunks", que consistem em fragmentar uma grande quantidade de informações em pequenos trechos que descrevem um determinado contexto, sendo também possível realizar essa fragmentação de maneira semântica e automática, a partir do uso de um modelo de embedding. Para garantir que a pergunta seja formulada de modo que a resposta correta esteja contida exclusivamente nesse contexto, sem depender de conhecimento externo, utilizou-se um LLM com alta capacidade de sintetizar perguntas e um outro com a função de verificar se o conteúdo está integralmente representado no texto de referência. Dessa maneira, é possível construir uma intrínseca relação entre contexto e pergunta que serve de entrada para os LLMs Geradores, permitindo avaliar a qualidade de suas respostas e estão sendo baseadas exclusivamente no texto fornecido como contexto, que, idealmente, na pipeline de RAG, será o texto recuperado pela consulta à base de conhecimentos.

Resposta dos LLMs Geradores (Modelos do MLC)

Depois de obter os pares de pergunta e contexto, o próximo passo é gerar as respostas de cada item utilizando um LLM Gerador. Para isso, foi empregada a biblioteca MLC-LLM (MLC-LLM, 2023), que compila diversos modelos de linguagem com compatibilidade para execução em um número considerável de plataformas, sendo uma delas, o celular. O uso desse *framework* possibilita a análise de diferentes modelos utilizando a mesma estrutura de contexto-pergunta, garantindo agilidade e diversidade para explorar diferentes modelos e hiper-parâmetros

Com base nisso, foi elaborado um *prompt* que descreve o contexto e a pergunta proposta, com o objetivo de extrair a informação desejada do texto. Nessa estrutura, foi imposta uma limitação para que a resposta fosse gerada apenas com base no texto de referência, evitando qualquer informação prévia ou conhecimento armazenado no LLM gerador. Para aumentar a complexidade do desafio e testar a robustez da extração de contexto na resolução do questionamento, além do par contexto-pergunta, foram introduzidos contextos extras no *prompt* que são desconexos à pergunta realizada. Isso permitiu observar se a resposta gerada pelo LLM se limitava ao contexto de referência, ignorando os textos externos, pois em uma pipeline de execução

de real de RAG, nem todos os contextos retornados serão corretamente referenciados à pergunta, tendo em vista que o assunto pode não estar presente na base de conhecimento. A quantidade de informações adicionais variou, permitindo um estudo da robustez do LLM Gerador diante do excesso de informações disponíveis.

Com essas informações em mãos, o próximo passo é gerar as respostas de cada LLM em estudo que estão disponíveis no MLC. Foram escolhidas famílias de modelos como Llama3, TinyLlama1, Phi3, Gemma, Mistral, Qwen2, Hermes2, entre outros. Para cada modelo, foi passado um *prompt* que, de forma geral, descreve a tarefa a ser realizada, faz referência ao *chunk* (ou conjunto de *chunks*) de entrada, e finaliza com a pergunta de interesse. Para avaliar a robustez de cada modelo e analisar suas variações, para cada modelo foi executado por um número de vezes o mesmo prompt, permitindo uma melhor avaliação de desempenho. Após essa etapa, o usuário se depara com uma grande quantidade de *chunks*, perguntas e respostas, tornando o processo de avaliação desses resultados bastante complexo. Por isso, a atividade de avaliação é crucial para auxiliar nesse processo.

Avaliação do LLM Juiz

A etapa final do fluxo de trabalho consiste na avaliação das respostas geradas por cada LLM. Como mencionado anteriormente, o volume de informações é extremamente elevado e escala de forma muito rápida, e a dependência de um observador humano para avaliar subjetivamente a qualidade de cada resposta (para repetições de um conjunto de perguntas para todos um grande conjunto de modelos) pode ser impraticável. Por isso, utilizou-se um outro LLM com alta capacidade para realizar a avaliação das respostas de maneira objetiva e com praticidade. De modo específico, esse modelo é aplicado para avaliar cada conjunto de *chunks*, perguntas e respostas geradas, analisando todo o contexto disponível e atribuindo uma nota que reflete a qualidade da resposta (Roucher, 2024). A Tabela 1 apresenta a forma como as notas atribuídas pelo LLM Juiz são definidas.

Tabela 1: Notas e suas respectivas descrições para avaliar as respostas geradas pelo LLM

Nota	Descrição
1	A resposta gerada informa que a informação não se encontra no contexto
2	A resposta é terrível: irrelevante ou parcialmente referente ao contexto

3	A resposta é ruim: despreza aspectos relevantes da questão
4	A resposta é boa: responde ao que se pede, porém pode ser melhorado
5	A resposta é ótima: responde de forma completa, detalhada e direta

Como resposta, nota-se que o LLM Juiz demonstrou boa capacidade de avaliar as respostas geradas pelos LLMs. Com base no contexto disponível a partir da junção de chunks, o juiz consegue descrever em detalhes as informações relevantes que a resposta gerada conseguiu extrair, destacando a clareza da informação e a facilidade de compreensão da resposta. Além disso, o LLM Juiz também aponta as falhas dos geradores que apresentaram desempenho inferior, como a afirmação de que não há informações úteis no texto para a resposta do usuário, quando, na verdade, essas informações estão bem descritas no chunk em análise. Quanto à nota gerada pelo LLM Juiz, o cálculo de métricas (média, desvio padrão, mediana, etc.) auxilia na filtragem dos modelos que respondem com maior qualidade em relação a outros, sendo bastante útil na construção de um ranking entre os LLMs Geradores.

Observações e Obstáculos Superados:

Neste contexto, é importante destacar os avanços realizados durante o processo de elaboração da *pipeline* para a avaliação de LLMs para execução em dispositivos móveis junto ao RAG. Em primeiro lugar, a construção de um *prompt* adequado foi um passo fundamental, pois permitiu que o LLM Juiz pudesse extrair todas as nuances das respostas geradas e avaliar corretamente (de maneira subjetiva) as informações, garantindo que elas fossem consistentes com o texto de referência para o usuário final. Além disso, atribuir uma nota para indicar a qualidade da resposta do LLM Gerador foi extremamente útil para simplificar e automatizar a filtragem dos modelos mais promissores, uma vez que a análise de métricas numéricas é mais fácil do que a leitura de todas as respostas geradas.

Por outro lado, um dos principais desafios encontrados foi na geração das respostas pelo LLM Gerador, já que a biblioteca do MLC ainda apresenta vários problemas de compilação e execução de determinados modelos. Para superar esse problema, foi necessário filtrar a família de modelos e suas respectivas quantizações que conseguiram rodar de forma satisfatória em dispositivos móveis. Dessa forma, foi possível realizar diversos experimentos para resolver a questão em estudo.

Como mencionado anteriormente, a quantidade de dados gerados a partir da metodologia completa aplicada é bastante elevada, o que tornaria inviável uma análise subjetiva humana. Nesse contexto, a automatização por meio de LLM para avaliar as respostas geradas mostrou-se uma ferramenta extremamente relevante para agilizar e reduzir os custos dessas operações.

Em relação ao LLM Juiz, observa-se que seu tempo de execução é maior que o dos outros modelos geradores, devido a sua complexidade computacional. Isso é relevante, pois, ao lidar com bases de dados maiores, essa operação pode se tornar consideravelmente custosa. Nesse sentido, a estrutura organizacional proposta neste trabalho viabiliza a realização do maior número possível de testes, possibilitando uma avaliação mais robusta dos LLMs Geradores.

Em resumo, observa-se que os LLMs Geradores mais recentes e com maior número de parâmetros treináveis demonstraram uma qualidade de resposta superior aos demais.

REFERÊNCIAS

Azaria, A., & Mitchell, T. (2023). *The internal state of an LLM knows when it's lying*. doi:10.48550/ARXIV.2304.13734

Es, S., James, J., Espinosa-Anke, L., & Schockaert, S. (2023). *RAGAS: Automated evaluation of Retrieval Augmented Generation*. doi:10.48550/ARXIV.2309.15217

Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M.-W. (2020). *REALM: Retrieval-Augmented Language Model pre-training*. doi:10.48550/ARXIV.2002.08909

Roucher, A. (2024). *Using LLM-as-a-judge for an automated and versatile evaluation.* HuggingFace. <https://huggingface.co/learn/cookbook/llm_judge>.

Lee, K., Chang, M.-W., & Toutanova, K. (2019). Latent retrieval for weakly supervised open domain question answering. doi:10.48550/ARXIV.1906.00300

Saad-Falcon, J., Khattab, O., Potts, C., & Zaharia, M. (2023). *ARES: An Automated evaluation framework for retrieval-augmented generation systems*. doi:10.48550/ARXIV.2311.09476

Team, M. L. C. (2023). MLC-LLM. Retrieved from https://github.com/mlc-ai/mlc-llm