

A Retrieval-Augmented Generation Information System for the Oil and Gas Industry

Rhuan Garcia de Assis Teixeira¹, Luciano Henrique Peixoto da Silva¹,
Thiago Oliveira-Santos¹, Alexandre Rodrigues², Marcos Pellegrini Ribeiro³,
Flávio Miguel Varejão¹

¹Departamento de Informática – Universidade Federal do Espírito Santo (UFES)
Caixa Postal 29075-910 – Vitória – ES – Brazil

²Departamento de Estatística – Universidade Federal do Espírito Santo (UFES)
Caixa Postal 29075-910 – Vitória – ES – Brazil

³Petróleo Brasileiro S.A.
Av. Horácio Macedo 950, Cid. Universitária - Ilha do Fundão
– Caixa Postal 21941-598 – Rio de Janeiro – RJ – Brazil

{rhuan.teixeira, luciano.h.silva}@edu.ufes.br,

{thiago.o.santos, alexandre.rodrigues, flavio.varejao}@ufes.br

mpellegrini@petrobras.com.br

Abstract. Research Context: The oil and gas industry relies on extensive and complex technical documentation, making manual information retrieval slow and inefficient for engineers and technicians who require quick and reliable answers for decisions. **Scientific and/or Practical Problem:** Generalist Large Language Models (LLMs) often struggle with specialized domains, leading to inaccuracies, contextual errors ("hallucinations"), and an inability to handle specific technical jargon. **Proposed Solution and/or Analysis:** This paper presents "RAG Petrolês" an information system assistant built using Retrieval-Augmented Generation (RAG) on "Petrolês", a collection of Portuguese theses and dissertations on the topic. It uses the IBM Granite model for embeddings, FAISS for similarity search, and a combination of the DeepSeek R1 and Mistral Small 3.2 LLMs to generate and refine answers. **Related IS Theory:** The work is grounded in the Information Processing theory where a Retrieval-Augmented Generation (RAG) system integrates external knowledge bases with LLMs to enhance accuracy, contextual relevance of their responses and reduce hallucinations. **Research Method:** A quantitative and qualitative evaluation was performed. The system's performance was tested against a custom dataset of 1500 questions. The evaluation involved a manual analysis of 150 answers and a broader statistical analysis of all 1500 responses using an "LLM-as-a-Judge" approach combined with Prediction-Powered Inference (PPI) to ensure robust results. **Summary of Results:** The proposed system achieved an accuracy of 88.88% in manual evaluation. The statistical analysis resulted in an accuracy range from 81.48% to 97.71% within a 95% confidence interval. Outperforming baseline models and demonstrating superior performance compared to a direct adaptation of a similar existing framework. **Contributions and Impact to IS area:** This research demonstrates the effectiveness of a specialized RAG system

in a technical, non-English domain. It provides a viable architecture for creating highly accurate, specialized information systems assistants without resorting to retrain foundational models.

1. Introdução

A indústria do petróleo e gás envolve processos altamente complexos, com demandas constantes por precisão, segurança e eficiência embasadas em um grande conhecimento técnico registrado em documentos extensos como normas, documentos técnicos, teses, dissertações e artigos científicos. Embora esses materiais sejam fontes valiosas de informação, o acesso manual a dados específicos em meio a milhares de páginas pode ser um processo lento e ineficiente. Engenheiros e técnicos, que dependem de informações precisas para orientar procedimentos críticos e decisões operacionais, necessitam de sistemas de informação capazes de fornecer respostas rápidas com informações confiáveis, reduzindo o tempo gasto na busca e interpretação de conteúdos acadêmicos e técnicos.

Nos últimos anos, a inteligência artificial generativa, em especial os modelos de linguagem de grande escala (LLMs) vem se consolidando como uma abordagem promissora para o tratamento de informações, auxílio no entendimento e absorção de assuntos mais complexos, e execução de tarefas textuais, proporcionado, por essas capacidades, novas formas de interação com bases de conhecimento, novas formas de sistemas de informação, sendo um bom exemplo a criação de assistentes inteligentes (chatbots[Dam et al. 2024]) que respondem questões e recuperam informações de maneira acessível e prática.

Apesar de seu potencial, LLMs por serem generalistas, apresentam limitações quando aplicadas a domínios especializados e a áreas técnicas como o domínio do conhecimento técnico e prático sobre indústria de Óleo e Gás, foco deste artigo, ou a um domínio de conhecimentos sobre uma dada região com datas e eventos como o abordado por [Sun et al. 2024]. Nesses casos, LLMs enfrentam desafios relacionados à precisão e contextualização das respostas geradas porque os modelos treinados de forma geral tendem a apresentar lacunas de conhecimento quando confrontados com perguntas muito específicas ou técnicas.

Nesse contexto, a técnica de *Retrieval Augmented Generation* (RAG)[Lewis et al. 2021] vem sendo explorada como uma solução viável para contornar essas limitações. A técnica consiste em integrar mecanismos de recuperação de informações que provenham contextos, trechos, relevantes para a pergunta a ser respondida a partir de uma base de conhecimento estruturada para que o modelo de linguagem produza uma resposta, reduzindo a ocorrência de alucinações e aumentando a precisão contextual. Dessa forma, torna-se possível adaptar modelos de linguagem de grande escala a domínios técnicos, aproximando suas respostas do conhecimento necessário para apoiar práticas profissionais.

Neste artigo, é apresentado o RAG Petrolês, um sistema que combina a técnica de Retrieval-Augmented Generation [Lewis et al. 2021] com o corpus especializado em óleo e gás em português Petrolês [Cordeiro 2020], que abrange o conhecimento técnico e prático da indústria de óleo e gás. O sistema foi desenvolvido a partir da segmentação do corpus em unidades menores, geração de representações numéricas (*embeddings*) por meio de um modelo otimizado para o português (Granite [Awasthy et al. 2025]),

indexação utilizando o FAISS e da geração de respostas por meio da combinação de modelos de linguagem de grande escala, especificamente o DeepSeek R1 [et al. 2025a] e o Mistral Small 3.2 [Mistral AI 2025].

O objetivo deste estudo é propor, implementar e avaliar um sistema de informação assistente especializado no domínio de óleo e gás, capaz de fornecer respostas precisas, contextualizadas e semanticamente adequadas a partir de um corpus técnico em português. Busca-se, com isso, demonstrar que a adoção de uma abordagem baseada em Retrieval-Augmented Generation, aliada a modelos de linguagem generalistas, é capaz de reduzir inconsistências e alucinações sobre um domínio específico, superando o desempenho do uso direto de LLMs generalistas e de abordagens similares descritas na literatura.

Como parte do processo de avaliação e teste do sistema, detalhado na secção 4, se fez necessário o desenvolvimento de um dataset de perguntas e respostas, para isso foram elaboradas manualmente um conjunto de 45 perguntas sobre o conhecimento presente no Petrolês que foi expandido a um total de 1500 perguntas com o uso de Gemma 3, um Large Language Model, o que resultou em um bom e extenso dataset de perguntas no domínio de óleo e gás.

A estrutura deste artigo está organizada da seguinte forma: a secção 2 apresenta os conceitos fundamentais sobre RAGs e o corpus Petrolês. A secção 3 detalha a arquitetura e o fluxo de trabalho do sistema RAG Petrolês. A secção 4 descreve a metodologia experimental, incluindo a criação do dataset e os métodos de avaliação. A secção 5 apresenta e analisa os resultados quantitativos e qualitativos obtidos. Por fim, a secção 6 conclui o trabalho, resumizando as principais descobertas.

2. RAGs e Petrolês

A construção de um sistema de informação assistente especializado em um domínio técnico exige uma compreensão clara das bases conceituais que sustentam a solução proposta. Nesta secção, são apresentados os elementos fundamentais que embasam o desenvolvimento do sistema: a técnica de Retrieval-Augmented Generation (RAG), utilizada para mitigar lacunas de conhecimentos e alucinações dos modelos de LLM; e o corpus Petrolês, que constitui a base de conhecimento adotada neste estudo para prover ao assistente sua especialização na indústria de petróleo e gás.

2.1. RAGs

A fim de implementar de forma adequada e adaptada a técnica de Retrieval-Augmented Generation [Lewis et al. 2021], objetivando maximizar os benefícios de menor alucinação e inferências mais factuais e interpretáveis [Lewis et al. 2021], foram analisadas algumas abordagens para a construção de assistentes especializados, que também utilizem RAG para seu objetivo de contornar as limitações de LLMs genéricas ao lidar com jargões e conhecimentos muito particulares de uma área.

Uma das abordagens encontradas foi a "Bailicai"[Long et al. 2024], que para resolver o problema da implementação de um assistente baseado em LLM e RAG especializado em um domínio médico de diagnósticos, propõe um framework específico ao domínio composto de 4 principais módulos. O primeiro sendo responsável por identificar se há ou não a necessidade de conhecimento externo para a query; O segundo responsável por decompor a query em um grafo direcionado acíclico de tarefas a serem executadas

pelo médico em um diagnóstico; O terceiro responsável por, caso necessário, resgatar os documentos relevantes no repositório e os rankear (parte em que realmente é implementado o retrieval); E por fim o quarto módulo em que a resposta é formulada integrando as informações resgatadas anteriormente das bases médicas para enriquecer as respostas e reduzir alucinações e ruídos.

Outra técnica, conhecida como *Retrieval Augmented Fine Tuning* (RAFT) [Zhang et al. 2024] sugere um ajuste fino do modelo para que aprenda a responder perguntas de um domínio específico consultando materiais externos. Durante esse treinamento o modelo é exposto tanto a documentos relevantes quanto "distratores", documentos irrelevantes, com o intuito de "ensinar" o modelo a ignorar informações não relevantes e citar a fonte correta da resposta.

[Sun et al. 2024] cria um sistema de informação assistente especializado em domínios específicos usando a técnica RAG. O sistema foi projetado para responder a perguntas sobre a Universidade Carnegie Mellon e a cidade de Pittsburgh, e sua estrutura se baseou na coleta de mais de 1.800 sub páginas da web, 7 PDFs e 16 tabelas relacionadas a esses tópicos. Os pesquisadores usaram uma combinação de anotação manual e automática para gerar 1.467 pares de perguntas e respostas para treinar e avaliar o sistema. Para materializar esse sistema, foi feito um pipeline de RAG que integra dois *retrievers* BM25[Chen and Wiseman 2023] e FAISS[Douze et al. 2025] utilizando all-MiniLM-L6-v2 [Wang et al. 2020] para *embedding*, com um módulo *re-ranker* que recebe os trechos resgatados para reordenar e evitar informações pouco relevantes entre os documentos recuperados por eles, e como LLM usado para a geração das respostas foi escolhido o Mistral 7B [Jiang et al. 2023]. Os resultados mostraram que o sistema RAG superou significativamente uma linha de base sem RAG, especialmente em consultas complexas e sensíveis ao tempo.

Assim, embora a técnica RAFT [Zhang et al. 2024] apresente resultados promissores ao adaptar modelos de linguagem a domínios específicos, sua aplicação exige um processo de treinamento supervisionado extenso. Etapa essa que requer recursos computacionais significativos, especialmente GPUs com alta capacidade de memória, o que excede as limitações de hardware do laboratório em que este trabalho foi desenvolvido. Além disso, por depender de treinamento, o RAFT vai além do escopo proposto, que busca avaliar abordagens viáveis sem re-treinar modelos de base e com foco na reutilização de LLMs já disponíveis. De forma semelhante, o framework Bailicai [Long et al. 2024], embora apresente uma arquitetura modular avançada, foi projetado especificamente para fluxos diagnósticos médicos, baseados em grafos de tarefas e etapas hierarquizadas, o que não se ajusta à natureza aberta e textual das consultas no domínio do petróleo e gás.

Logo se destaca entre essas o trabalho desenvolvido por [Sun et al. 2024] devido a semelhança ao problema abordado neste trabalho, focando na construção um sistema de informação assistente com o conhecimento de um corpus técnico especializado utilizando RAG e uma LLM genérica, sem requerer extensivo treinamento do modelo como a abordagem usada em RAFT [Zhang et al. 2024] e sem ter que lidar com a construção de frameworks e módulos específicos para a tarefa a ser melhorada, tal como a abordagem usada no "Bailicai"[Long et al. 2024] para a tarefa de diagnóstico médico. Outra vantagem dessa abordagem para as demais é manter a generalidade do assistente enquanto

obtem um bom desempenho sob o domínio de interesse.

Por esses motivos foi optado neste trabalho, a fim de obter um bom sistema referência para comparação, fazer uma adaptação do sistema descrito por [Sun et al. 2024] ao domínio de óleo e gás, utilizando do mesmo corpus de nossa abordagem e ajustando hiperparâmetros de pré-processamento (como *chunk-size* e *overlap*) iguais aos utilizados pela proposta deste artigo. Adaptação essa que será melhor descrita quando relevante.

2.2. Petrolês

A fim de construirmos o assistente especializado no domínio técnico da indústria do petróleo e gás foi necessário usar uma base ampla de conhecimento confiável e representativo do domínio. Embora fosse possível construir um novo corpus textual sobre o tema do zero, a fim de não fugir do escopo principal de desenvolver um sistema de Retrieval-Augmented Generation (RAG), buscou-se uma base de dados já existente e estabelecida. Para esse propósito foi utilizado o Petrolês [Cordeiro 2020], um corpus público formado por teses e dissertações em português, no domínio de óleo e gás, disponibilizado publicamente em <https://www.petroles.ica.ele.puc-rio.br>.

O Petrolês é um corpus textual obtido através da compilação de teses e dissertações relacionadas a área de petróleo e gás da Biblioteca Digital Brasileira de Teses e Dissertações através de uma seleção feita aplicando técnicas de aprendizado de máquina para filtrar os documentos mais relevantes ao domínio. Assim, foi construído esse corpus que contém cerca de 6 milhões de sentenças e 74,4 milhões de *tokens* [Cordeiro 2020].

Durante o pré-processamento do Petrolês foram adotadas medidas para remover ruídos comuns quando há a extração de texto de um PDF, o que resultou na remoção das quebras de página, formatações não textuais e sequências numéricas objetivando remover ruídos como a numeração das páginas e de seções dentro de um artigo. A remoção de dados numéricos do corpus contribuiu para a geração de um corpus mais uniforme, porém também provocou a perda de diversos dados numéricos relevantes dos documentos, o que acaba sendo um problema para o sistema de informação assistente.

3. O RAG Petrolês

Esta seção detalha a arquitetura e o processo de implementação do sistema RAG Petrolês. O objetivo é detalhar como a técnica RAG foi adaptada a partir do trabalho de [Sun et al. 2024] para a construção do sistema de informação assistente para o domínio específico do Petrolês.

Esta seção foca ainda a descrição em duas etapas fundamentais para desenvolvimento da abordagem proposta neste trabalho: o processo de criação do banco de vetores a partir do corpus Petrolês; e o fluxo operacional do sistema que explica como uma pergunta do usuário é processada, desde a recuperação dos contextos relevantes até a geração da resposta final pela combinação de LLMs.

3.1. Criação do banco de vetores

Com base no framework desenvolvido em [Sun et al. 2024], o corpus utilizado foi dividido em chunks menores utilizando uma funcionalidade disponibilizada pela biblioteca LangChain [Chase et al. 2022] para partir um grande texto em pedaços menores tentando

preservar a lógica natural com base em uma hierarquia de separadores (parágrafos, frases, palavras e caracteres). Isso foi feito a fim de transformar o grande corpus em pequenos trechos de tamanho próximo a 1024 caracteres, tornando assim o corpus manejável para o processamento e uso com os modelos disponíveis no mercado atual, já que caso contrário, pelo grande tamanho do corpus não seria possível o encaixe nas janelas de contexto dos modelos. A fim de garantir uma menor perda de informações nesse processo de divisão, também foi definida uma sobreposição no tamanho de 250 caracteres, dando uma maior margem para a função de divisão poder preservar pedaços maiores.

Esses trechos (chunks) resultantes são processados por um modelo de *embedding*, sendo o escolhido nesse projeto o IBM Granite Embedding [Awasthy et al. 2025] com 278 milhões de parâmetros por ter sido o que melhor lidou, entre as opções disponíveis, com os textos em português brasileiro e o jargão utilizado no domínio. Esse modelo é responsável por mapear o texto do trecho a uma representação numérica vetorial de dimensionalidade menor, mais densa, que mantenha um significado semântico, facilitando a compreensão de relações entre textos, sendo essa representação vetorial (*embedding*) que posteriormente será utilizada para calcular similaridades semânticas entre os trechos.

Os *embeddings* gerados são utilizados para indexar os trechos no FAISS[Douze et al. 2025], uma biblioteca de código aberto para armazenamento e busca eficiente de similaridade em conjuntos de vetores de alta dimensionalidade (em nosso caso o banco de *embeddings*) baseando-se no algoritmo de Approximate Nearest Neighbors [Fu et al. 2025] e clustering desses vetores [Dong et al. 2011], otimizando esse processo com o uso de GPU[Johnson et al. 2017] e estruturas de indexação que tiram proveito de quantização[Guo et al. 2020] e Hierarchical Navigable Small World [Malkov and Yashunin 2018]. Isso resulta em um *retriever* (recuperador) FAISS [Douze et al. 2025] construído com os *chunks* (trechos) do petrolês que nos possibilita buscar e recuperar rapidamente trechos semanticamente similares a outro, de grande utilidade no workflow do RAG.

3.2. Workflow do RAG

Como podemos ver na figura 1, o sistema inicialmente faz uma busca, utilizando o *retriever* previamente construído, dos trechos do Petrolês com semântica mais próxima a da pergunta feita pelo usuário. Após isso, é passado à LLM a pergunta do usuário e os top-k trechos mais relacionados à pergunta obtidos pela pesquisa (neste trabalho, k=5) através de um prompt elaborado de forma a permitir e incentivar o uso do conhecimento prévio da LLM combinado aos trechos recuperados, se restringir a esse contexto, gerar respostas claras e amigáveis em português e se limitar em relação a números (devido a ausência no petrolês dos dados numéricos). A resposta dessa LLM é então apresentada ao usuário, sendo a saída do sistema.

O prompt base referido, no qual o sistema insere os trechos resgatados e a pergunta do usuário a ser respondida está disponível no Apêndice A.

Por ter se mostrado um modelo adequado a tarefa, a LLM escolhida para gerar a resposta ao usuário foi o DeepSeek R1 [et al. 2025a] devido a sua capacidade de raciocínio, paradigma esse que “permite LLMs a imitar complexos processos de pensamento humano, como busca em árvore e pensamento reflexivo” [Xu et al. 2025]. Porém, para contornar um problema ocasional dessa LLM, relatado no artigo do DeepSeek R1,

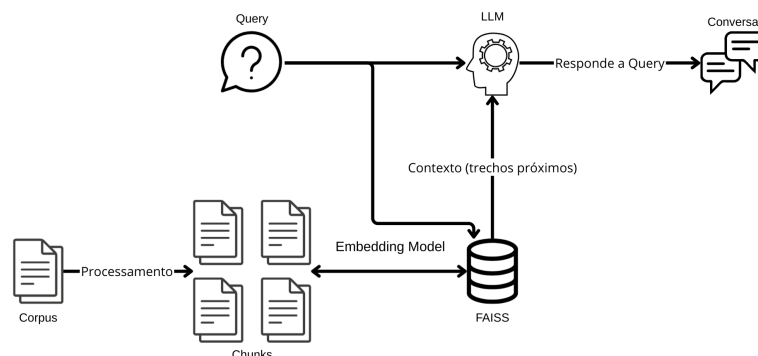


Figura 1. Workflow do RAG desenvolvido

de mistura de linguagens especialmente quando trabalhando em linguagens diferentes de chinês e inglês, foi utilizado um segundo modelo Mistral Small 3.2 [Mistral AI 2025], que recebe a resposta obtida do DeepSeek em um prompt com ordens para corrigir erros ortográficos, mistura de linguagens e traduzir para português caso necessário, gerando a resposta final apresentada ao usuário. Essa mistura de linguagens descrita acima pode ser observada no exemplo de resposta gerada sem o uso do segundo modelo para correção presente na tabela 1.

Pergunta: Como podemos classificar ignição de uma combustão polimérica?

Resposta DeepSeek R1: A ignição de uma combustão polimérica pode ser classificada considerando several aspects:

1. Tipo de Ignição:

- **Ignição Espontânea:** Acontece quando a liberação de 热量 durante a decomposição do polímero é suficiente para atingir a temperatura de autoignição, mantendo a combustão sem uma fonte externa de calor.
- **Ignição não Espontânea:** Requer uma fonte externa de calor ou chama, como um bico de Bunsen, para iniciar o processo.

Tabela 1. Mistura de linguagens feita pelo melelo DeepSeek R1

Isso resulta em um sistema que aplica Retrieval-Augmented-Generation [Lewis et al. 2021] em uma implementação similar a proposta por [Sun et al. 2024]. Entretanto, apresenta diferenças notáveis em relação a esses trabalhos. Entre elas, destaca-se o uso reduzido da ferramenta Langchain [Chase et al. 2022], o que contribui para maior escalabilidade. Além disso, o sistema faz uso de modelos mais recentes, que incorporam técnicas avançadas de treinamento e otimização como o uso de *reasoning* [Xu et al. 2025] feito pelo modelo de LLM escolhido para gerar as respostas, o que se alia ao modelo de *embedding* mais atual e otimizado à linguagem utilizado para *retrieval* em no sistema, resultando em um fornecimento ao modelo de informações relevantes mais frequentemente.

A fins de comparação, também foi implementado sob os mesmos *chunks* de aproximadamente 1024 caracteres com *overlap* de 250 caracteres, o pipeline

(workflow) descrito em [Sun et al. 2024] utilizando dois métodos de retrieval, FAISS [Douze et al. 2025], utilizando all-MiniLM-L6-v2 [Wang et al. 2020] para embedding, objetivando uma rápida busca sob um grande volume de documentos sacrificando alguma acurácia, e BM25 [Chen and Wiseman 2023] objetivando compensar essa perda de acurácia e melhorar a precisão dos documentos resgatados. Esses documentos resgatados são então passados por um módulo de re-ranking a fim de minimizar a perda de performance do RAG vinda de documentos não relacionados estarem sendo passados a LLM e enfim são entrada do modelo de LLM escolhido (Mistral 7B [Jiang et al. 2023]) através de um *prompt* elaborado para gerar a resposta da pergunta. Resultando em um assistente especializado no nosso domínio, construído em uma adaptação do sistema proposto por [Sun et al. 2024] que será utilizado como uma referência de comparação neste artigo.

4. Metodologia de experimentação

Para avaliar o sistema de forma similar ao uso comum imaginado (*question-answering* de perguntas sobre a área de petróleo e gás), foi elaborado sobre o domínio do petróleo um dataset de 1500 pares de perguntas e trechos do corpus que as originaram. Essas perguntas foram então respondidas pelo sistema de RAG proposto, caracterizado pela resposta baseada em um contexto resgatado pelo FAISS, gerada pelo DeepSeek R1[et al. 2025a] e traduzida pelo Mistral Small 3.2[Mistral AI 2025] e por outras três abordagens, utilizadas para comparação:

- a) uso da LLM DeepSeek R1 14B[et al. 2025a] com o auxílio do modelo Mistral Small 3.2 [Mistral AI 2025] sem o uso de RAG (DeepSeek R1 + Mistral);
- b) uso da LLM Mistral 7B [Jiang et al. 2023] sem o uso de RAG (Mistral 7B);
- c) uso de RAG seguindo a metodologia proposta em [Sun et al. 2024], com um pipeline desenvolvido com [Chase et al. 2022], composto por dois *retrievers*: FAISS utilizando All-MiniLM-L6-v2[Wang et al. 2020] para geração dos embeddings e BM25, responsáveis por fornecer um contexto a LLM Mistral 7B [Jiang et al. 2023](Mistral 7B com retrieval FAISS + BM25).

As respostas fornecidas pelas diferentes abordagens foram avaliadas, e realizou-se uma comparação quantitativa e qualitativa sob elas. É Importante destacar que os resultados dos experimentos devem trazer respostas para três questionamentos principais:

- Quanto conhecimento o sistema final demonstra sobre o corpus de domínio específico?
- Como se compara com as abordagens *baseline*, sem o uso de RAG?
- Como se compara com a adaptação direta do trabalho de [Sun et al. 2024], dadas as similaridades dos problemas e abordagens?

A geração das perguntas e respostas para todas as 1500 questões seria um processo muito laboroso e impraticável no contexto em que foi realizado este trabalho. Para contornar esse problema, foi elaborada uma metodologia semi-automática para construção do dataset e para a avaliação das respostas dadas pelas diferentes abordagens. As duas subseções seguintes explicam como o dataset de perguntas foi elaborado e como as respostas das diferentes abordagens foi avaliada semi-automaticamente.

4.1. Obtenção do Dataset de Perguntas

Inicialmente foram elaboradas manualmente 50 perguntas que foram pareadas com trechos do Petrolês que contém sua resposta. A partir dessas perguntas iniciais, foi utilizado o modelo Gemma 3 [et al. 2025b], em sua variação de 7 bilhões de parâmetros, para gerar as demais 1450 perguntas. Esse processo foi realizado através de *prompts* contendo um trecho selecionado aleatoriamente do petrolês, acompanhado de seu trecho antecedente e subsequente (a fim de evitar problemas com contextos interrompidos pela divisão de chunks), 3 exemplos das perguntas anteriormente desenvolvidas e a instrução para elaborar, sobre o trecho do corpus selecionado, uma nova questão. Dentre essas intruções estavam observações para se manter o nível de dificuldade e estrutura similar aos exemplos passados (compondo então uma aplicação de *few-shot-prompting* [et al. 2020]).

Esse *prompt* também possui guias para a elaboração de perguntas a fim de contornar os problemas do corpus em que estamos trabalhando, como ignorar anotações de formatação, apenas elaborar questões que possam ser respondidas com o trecho passado e não formular questões relacionadas a números específicos, já que o corpus não possui dados numéricos. Assim foi construído um dataset de 1500 perguntas contendo para cada pergunta os trechos que originaram a mesma e que consequentemente devem conter sua resposta.

4.2. Avaliação das Respostas

Como mencionado anteriormente, a avaliação manual das respostas obtidas pelas 4 abordagens para todas as 1500 perguntas do dataset foi impraticável. Para simplificar esse processo, foi adotada uma metodologia de avaliação realizada em três etapas complementares:

A primeira etapa envolve a avaliação manual [Yu et al. 2025] das respostas de um subconjunto de 150 perguntas por cada modelo. As respostas de cada modelo para essas perguntas foram corrigidas manualmente por um ser humano e classificadas como corretas ou incorretas com base nos seguintes critérios:

- **Verificação de fatos:** Respostas com informações incorretas são consideradas inteiramente incorretas.
- **Coerência com a pergunta:** Respostas que não abordam diretamente a pergunta são consideradas incorretas.
- **Adesão à linguagem:** Respostas que desviam da linguagem utilizada (em nosso caso, Português) são consideradas incorretas.
- **Erros ortográficos e gramaticais:** Respostas com erros significativos de linguagem ou que dificultem a compreensão do usuário são consideradas incorretas.
- **Compleitude da resposta:** Respostas que não fornecem todas as informações necessárias ou deixam lacunas significativas são consideradas incorretas.
- **Clareza e concisão:** Respostas prolixas ou difíceis de entender são consideradas incorretas.
- **Relevância da informação:** Respostas que incluem informações irrelevantes ou desnecessárias são consideradas incorretas.

Cabe destacar que somente se a resposta for enquadrada como correta por todos os critérios que ela é considerada correta para o cálculo da métrica de avaliação, no caso deste trabalho, a acurácia.

A segunda etapa envolve a avaliação de todas as respostas de cada modelo para as 1500 perguntas usando uma abordagem na qual se usa uma LLM como verificador da correção das respostas. Essa abordagem é conhecida como a técnica de *LLM-as-a-Judge* [Gu et al. 2025]. Neste trabalho as respostas foram avaliadas pelo modelo de linguagem Llama 3.3 24B [Meta Platforms, Inc. 2025] utilizando, para melhor desempenho do modelo, *few-shot-prompting* [et al. 2020] (provendo no *prompt* 3 exemplos de respostas corretas e 2 exemplos de incorretas) e o trecho do petrolês que originou a pergunta (e consequentemente terá a informação da resposta correta).

A terceira etapa envolve o uso da técnica de Prediction-Powered Inference (PPI) [Angelopoulos et al. 2023]. Ela utiliza as avaliações feitas pela LLM para a aplicação de uma estrutura estatística que permite combinar um pequeno conjunto de dados rotulados de alta qualidade (como o nosso conjunto de dados avaliados manualmente), chamada de *gold-standard*, a um conjunto de dados não rotulados acompanhados de previsões de um modelo de aprendizado de máquina (como as questões corrigidas apenas pelo *LLM-as-a-Judge*) a fim de construir intervalos de confiança estatisticamente válidos mas com menor margem de erro quando comparado ao intervalo baseado somente no conjunto rotulado. Para fazer isso, o PPI utiliza um *rectifier* (retificador) que mede o erro sistemático das previsões (providas pela LLM de correção de questões) em relação aos dados reais. Esse retificador é estimado a partir do pequeno conjunto de dados rotulados de alta qualidade comparando o dado confiável e sua previsão atribuída e então é utilizado para corrigir o viés dos resultados obtidos com base apenas nas correções do LLM.

Dessa forma, foi possível fazer uma avaliação quantitativa adequada dentro das limitações de recursos disponíveis.

5. Resultados e análise

Nesta secção, são apresentados e analisados os resultados obtidos da aplicação da metodologia descrita acima. A análise inicia-se pelo subset de perguntas avaliadas manualmente e, em seguida, é expandida ao resultado obtido com o uso do Prediction Powered Inference [Angelopoulos et al. 2023]. Essa análise inicial é então seguida por uma avaliação detalhada dos resultados, respondendo as perguntas de pesquisa levantadas anteriormente e fazendo observações qualitativas sobre as respostas dos sistemas analisados.

5.1. Avaliação Manual

A Tabela 2 apresenta o número de respostas corretas de cada abordagem das 150 perguntas selecionadas para correção manual e a respectiva acurácia.

Abordagem	Respostas Corretas	Acurácia (%)
Mistral 7B	83	54.25
Mistral 7B com retrieval FAISS + BM25 (Sun)	92	60.13
DeepSeek R1 + Mistral	96	62.75
DeepSeek R1 + Mistral com retrieval FAISS (RAG Petrolês)	136	88.89

Tabela 2. Desempenho das abordagens - Avaliação manual

Observando os resultados é possível notar que o sistema RAG Petrolês teve uma performance consideravelmente superior aos demais comparados. Destaca-se principalmente a diferença entre a abordagem proposta em [Sun et al. 2024] e o RAG Petrolês. Essa diferença, entretanto, provavelmente é fundada e agravada pela desigualdade entre os modelos base (Mistral 7B e DeepSeek R1 + Mistral) resultado de fatores como a diferença no treinamento, quantidade de parâmetros e capacidade de *reasoning* presente no DeepSeek R1.

Algo que também deve ser levado em consideração é o modelo de *embedding* utilizado. Nesse quesito a abordagem de [Sun et al. 2024], por utilizar um modelo sem grande enfoque em suporte de múltiplas linguas entra em desvantagem em uma base de conhecimento em português. Fazendo com que resgate trechos relevantes menos consistentemente, apesar do uso de BM25[Chen and Wiseman 2023] adicional.

5.2. PPI - Prediction Powered Inference

A Tabela 3 apresenta a acurácia alcançada e o correspondente intervalo de confiança com grau de 95% na avaliação das respostas das 1500 perguntas de cada modelo.

Sistema avaliado	Acurácia (%) com $IC_{95\%}$
Mistral 7B	44.81 (31.71 - 57.73)
Mistral 7B com retrieval FAISS + BM25 (Sun)	62.99 (50.63 - 74.89)
DeepSeek R1 + Mistral	64.94 (48.63 - 73.27)
DeepSeek R1 + Mistral com retrieval FAISS (RAG Petrolês)	90.74 (81.48 - 97.71)

Tabela 3. Desempenho dos sistemas - Avaliação PPI

Como descrito anteriormente, o uso da estrutura estatística Prediction Powered Inference [Angelopoulos et al. 2023] foi motivado a obter um intervalo de confiança com menor margem de erro e certificar que não teríamos obtido apenas uma amostragem enviesada em meio as 1500 perguntas por exemplo.

A partir desses resultados é possível observar que há uma intersecção entre os intervalos de confiança das abordagens Mistral 7B, Sun e DeepSeek R1 + Mistral; indicando que são estatisticamente equivalentes. Já em relação ao RAG Petrolês, o limite inferior do seu intervalo de confiança é superior ao limite superior dos intervalos das demais abordagens, sugerindo assim que o RAG Petrolês apresentou desempenho estatisticamente superior nesse conjunto de testes.

Além disso, com esse intervalo de confiança podemos ver uma grande similaridade na acurácia do sistema proposto em [Sun et al. 2024] e o uso apenas do DeepSeek R1 14B (sem uso de Retrieval-Augmented-Generation) com Mistral Small 3.2 para tradução e coerência linguística. Isso se deve principalmente ao modelo utilizado para a geração das respostas, se tratando de um modelo maior (em quantidade de parâmetros), capaz de *reasoning* e treinado em uma base diferente de conhecimento.

5.3. Análise

Os resultados reportados possibilitam responder as questões de pesquisa formuladas na seção 4:

- **Qual o nível de conhecimento do sistema final sobre o corpus de domínio específico?** O sistema final demonstrou um conhecimento significativo sobre o corpus Petrolês, apresentando uma acurácia média na faixa de 90%.
- **Como ele se compara com o baseline, sem o uso de Retrieval-Augmented Generation?** O uso do RAG proporcionou um ganho de desempenho ao sistema em comparação com o uso direto do modelo, indo de 96 questões acertadas à 136 questões acertadas (um aumento percentual de 29%). Fazendo uso de mais conhecimentos e termos específicos, além de apresentar menos alucinações.
- **Como ele se compara com a adaptação direta do trabalho de [Sun et al. 2024], dadas as similaridades dos problemas?** Apesar das similaridades dos problemas e das soluções apresentadas, a abordagem desenvolvida neste trabalho obteve um desempenho significativamente superior, um aumento percentual de 32%, ao da abordagem que segue estritamente o trabalho de [Sun et al. 2024].

Essa grande diferença é possivelmente decorrente da desigualdade nos modelos de LLM utilizados como base, o que pode ser visto na diferença de acurácia entre os dois sistemas sem retrieval (Mistral 7B e DeepSeek R1 + Mistral). Além dessa diferença, o *retrieval* do RAG na implementação proposta utiliza um modelo de embedding mais avançado e adequado à língua portuguesa, algo que não foi enfrentado na implementação de [Sun et al. 2024] e fez com que o contexto fornecido ao modelo frequentemente possuísse mais informações relevantes e evitou que informações desconexas ao problema a ser resolvido fossem fornecidas à LLM.

Além das métricas quantitativas, foram observadas tendências e comportamentos nas respostas que, embora não mensuráveis numericamente, são cruciais para a implementação de assistentes com propósitos semelhantes.

Uma diferença notável reside no tempo de inferência entre os modelos Mistral 7B[Jiang et al. 2023] e DeepSeek R1 14B[et al. 2025a], especialmente na etapa de *reasoning*, na qual uma linha de pensamento é estabelecida antes da resposta final. Enquanto o Mistral infere uma resposta em aproximadamente 5 segundos, o DeepSeek, em conjunto com o Mistral para correção de erros de linguagem, leva cerca de 30 segundos para a mesma tarefa.

Há também uma distinção clara no formato das respostas. Os sistemas baseados no DeepSeek R1 frequentemente produzem saídas formatadas em Markdown, utilizando negrito para termos relevantes e tópicos para organizar a informação. Em contraste, os sistemas baseados no Mistral tendem a gerar textos simples e corridos. Essa diferença pode ser notada no exemplo presente no 4:

Pergunta: Quais componentes foram adicionados ao meio mineral para a produção de PHAs, além do glicerol e do NH_4^+ ?

Resposta DeepSeek R1: Os seguintes componentes foram adicionados ao meio mineral para a produção de PHAs além do glicerol e do NH_4^+ :

- **Fosfatos:** Na_2HPO_4 , KH_2PO_4
- **Minerais:** MgSO_4 , CaCl_2
- **Citrato férrico amoniacal**
- **Solução de elementos traços:** H_3BO_3 (boro), CoCl_2 (cobalto), ZnSO_4 (zinc), MnCl_2 (manganês), NaMoO_4 (molibdênio), NiCl_2 (nickel), CuSO_4 (cobre)

Estes componentes fornecem os nutrientes essenciais para o cultivo bacteriano e a síntese de PHAs.

Resposta Mistral: Além do glicerol e do NH_4 , o meio mineral utilizado na produção de PHAs também contém sais de sódio, potássio, magnésio, cálcio, ferro amoniacal, bório, cobalto, zinco, manganês, molibdênio, níquel, cobre, alumínio e um meio mineral de Bushnell-Haas.

Tabela 4. Comparação entre as respostas dos modelos DeepSeek R1 e Mistral

Consequentemente, os sistemas que utilizam o DeepSeek R1 apresentam uma menor frequência de citações diretas ou cópias literais de trechos do contexto fornecido. Eles demonstram uma maior capacidade de adaptar o contexto à pergunta, ao contrário dos sistemas baseados apenas no Mistral, que, embora forneçam trechos contextualmente adequados, muitas vezes respondem à pergunta de forma indireta.

É importante notar que há uma diferença de escalabilidade entre os frameworks de RAG testados, devido ao uso de LangChain [Chase et al. 2022] no framework proposto por Sun [Sun et al. 2024] que apesar de lidar bem com modelos de embedding menores, quando testado com modelos diferentes de embedding apresentou instabilidade ao lidar com os aproximadamente 535000 chunks com vetores de 768 floats (como o Granite utilizado na implementação proposta). Na arquitetura proposta neste artigo esse problema não foi observado pois o LangChain foi empregado apenas para a separação dos chunks.

6. Conclusões

Este trabalho demonstrou a viabilidade e a eficácia de um sistema de RAG na implementação de um sistema de informação assistente especializado sobre o domínio petrolífero, utilizando o corpus Petrolês. Os resultados obtidos destacam a capacidade do sistema em fornecer respostas precisas e contextualmente relevantes, superando significativamente as abordagens que não empregam RAG e apresentando melhor desempenho em comparação com adaptação direta de trabalho semelhante descrito na literatura.

A arquitetura proposta, que integra modelos recentes e avançados de linguagem, e um sistema de recuperação de informações otimizado, valida a hipótese de que a combinação estratégica de LLMs genéricos com bases de conhecimento específicas pode gerar sistemas de informação assistentes altamente especializados, sem a necessidade de extensivo treinamento de modelos de base ou o uso de hardware complexo.

Uma contribuição importante deste trabalho é a metodologia de experimentação desenvolvida, utilizando diferentes LLMs para definir perguntas, gerar respostas e corrigir as respostas dadas pelo sistema. Em particular, a integração das técnicas de LLM-as-

a-Judge e Prediction-Powered Inference se mostrou uma forma efetiva de avaliar de forma mais objetiva o desempenho das diferentes abordagens avaliadas neste trabalho.

Uma limitação do sistema desenvolvido é decorrente da forma que o corpus Petrolês foi elaborado, retirando-se muitos dos números e tabelas presentes nos artigos. Sem essas informações, o assistente possui dificuldade em responder questões dependentes de dados numéricos.

Outro problema observado é que ocasionalmente o corpus inteiro é interpretado como um artigo, levando a algumas respostas inesperadas quando a pergunta é pouco específica. Esse problema provavelmente é causado pela versão do corpus que foi utilizada, em que não há separações claras de início e final entre artigos. Devido a essa característica, durante o processamento não foi optado por gravar rótulos indicando origem de cada trecho (chunk), o que impossibilita posteriormente indicar no prompt ao modelo de LLM de onde cada trecho se origina e distinguir origens diferentes algo possivelmente mais efetivo que informar que se trata de um conjunto de artigos (como é feito na implementação atual).

Agradecimentos

OS autores agradecem o suporte fornecido por: CENPES-Petrobras, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001, Conselho Nacional de Desenvolvimento Científico e Tecnológico – Brasil (CNPq) e Fundação de Amparo à Pesquisa do Espírito Santo (FAPES, Brazil, 2022-NGKM5).

Referências

- Angelopoulos, A. N., Bates, S., Fannjiang, C., Jordan, M. I., and Zrnic, T. (2023). Prediction-powered inference.
- Awasthy, P., Trivedi, A., Li, Y., Bornea, M., Cox, D., Daniels, A., Franz, M., Goodhart, G., Iyer, B., Kumar, V., Lastras, L., McCarley, S., Murthy, R., P, V., Rosenthal, S., Roukos, S., Sen, J., Sharma, S., Sil, A., Soule, K., Sultan, A., and Florian, R. (2025). Granite embedding models.
- Chase, H. et al. (2022). Langchain: Building applications with llms through composability. <https://github.com/langchain-ai/langchain>. Acesso em: 15 de jul. de 2025.
- Chen, X. and Wiseman, S. (2023). Bm25 query augmentation learned end-to-end.
- Cordeiro, F. C. (2020). Petrolês – como construir um corpus especializado em óleo e gás em português. Master's thesis, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, Brasil.
- Dam, S. K., Hong, C. S., Qiao, Y., and Zhang, C. (2024). A complete survey on llm-based ai chatbots.
- Dong, W., Moses, C., and Li, K. (2011). Efficient k-nearest neighbor graph construction for generic similarity measures. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, page 577–586, New York, NY, USA. Association for Computing Machinery.

- Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.-E., Lomeli, M., Hosseini, L., and Jégou, H. (2025). The faiss library.
- et al., D.-A. (2025a). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning.
- et al., G. T. (2025b). Gemma 3 technical report.
- et al., T. B. B. (2020). Language models are few-shot learners.
- Fu, C., Xiang, C., Wang, C., and Cai, D. (2025). Fast approximate nearest neighbor search with the navigating spreading-out graph.
- Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K., Wang, Y., Gao, W., Ni, L., and Guo, J. (2025). A survey on llm-as-a-judge.
- Guo, R., Sun, P., Lindgren, E., Geng, Q., Simcha, D., Chern, F., and Kumar, S. (2020). Accelerating large-scale inference with anisotropic vector quantization.
- Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. (2023). Mistral 7b.
- Johnson, J., Douze, M., and Jégou, H. (2017). Billion-scale similarity search with gpus.
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S., and Kiela, D. (2021). Retrieval-augmented generation for knowledge-intensive nlp tasks.
- Long, C., Liu, Y., Ouyang, C., and Yu, Y. (2024). Bailicai: A domain-optimized retrieval-augmented generation framework for medical applications.
- Malkov, Y. A. and Yashunin, D. A. (2018). Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs.
- Meta Platforms, Inc. (2025). Meta llama 3.3. https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_3/. Acesso em: 15 de jul. de 2025.
- Mistral AI (2025). Mistral small 3.2. <https://huggingface.co/mistralai/Mistral-Small-3.2-24B-Instruct-2506>. Acessado em: 26 de setembro de 2025.
- Sun, H., Wang, Y., and Zhang, S. (2024). Retrieval-augmented generation for domain-specific question answering: A case study on pittsburgh and cmu.
- Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers.
- Xu, F., Hao, Q., Zong, Z., Wang, J., Zhang, Y., Wang, J., Lan, X., Gong, J., Ouyang, T., Meng, F., Shao, C., Yan, Y., Yang, Q., Song, Y., Ren, S., Hu, X., Li, Y., Feng, J., Gao, C., and Li, Y. (2025). Towards large reasoning models: A survey of reinforced reasoning with large language models.
- Yu, H., Gan, A., Zhang, K., Tong, S., Liu, Q., and Liu, Z. (2025). *Evaluation of Retrieval-Augmented Generation: A Survey*, page 102–120. Springer Nature, Singapore.

Zhang, T., Patil, S. G., Jain, N., Shen, S., Zaharia, M., Stoica, I., and Gonzalez, J. E.
(2024). Raft: Adapting language model to domain specific rag.

A. Apêndice A - Prompt Template

Listing 1. Prompt do Assistente Virtual

Você é um assistente virtual especializado em petróleo e para seu auxílio, sempre será fornecido um contexto contendo informações ligadas ao tema.

Responda às perguntas dos usuários com base em seu conhecimento e no contexto fornecido, de forma natural, explicativa e profissional, sem mencionar ou referir-se ao contexto explicitamente.

Diretrizes:

Descrição do Contexto: Nunca mencione que está usando um contexto ou cite elementos como textos, figuras, tabelas ou seções. O usuário deve pensar que o conhecimento partiu de você.

Idioma e Fluidez: Responda sempre em português, de maneira clara e amigável, como se você possuísse o conhecimento.

Restrições de Números:

Se a pergunta tiver partes envolvendo números, responda apenas as partes não numéricas.

Caso a resposta dependa exclusivamente de números, diga: "Não tenho a informação necessária porque a base de dados não trabalha com números."

Repare que na base de dados, não temos números, então se atente para nunca deixar a saída possuir <NUMBER>!

Incapacidade de Resposta:

Se o contexto não contiver a informação, responda: "Não tenho a informação necessária para responder à sua pergunta."

Proibição de Frases Introduzindo Respostas: Não inicie respostas com frases como "De acordo com", "A resposta é", ou similares.

Formato de Entrada:

Contexto: Bloco de texto com informações sobre petróleo.

Pergunta do Usuário: A questão a ser respondida.

Formato de Resposta:

Responda de forma clara, natural e informativa, usando apenas o contexto. Não mencione números se eles não estiverem no contexto.

--- Entrada

Contexto:

{text_books}

Pergunta do usuário: {message}